مجلة تقنية شهرية

مجلة الرابط

تقرؤون في هذا العدد:

- ليست رابطا بل انصهارا
- ثورة الخدمات: WCF نموذجا (الجزء الثاني)
- قصة ظهور قاصمة الظهور OOP (الجزء الثاني)
- الفرق بين Abstract Classes وبين Interfaces
- جولات في أعماق PHP (الفرق بين GET و POST)
- لست طبيبا أنا مريض مثلكم
- كيف برمجت أول لعبة أندرويد
- سبيل النجاح
- السلسلة التعليمية في التطور التكنولوجي
- التفكير البرمجي
- أدبيات فن تصميم المواقع
- أيقظها فإن عادت للنوم فنم
- قهوة الرابط

ومنافع أخرى

رئيس التحرير: خالد السعداني

مجلة الرابط

العدد الثاني - شهر أكتوبر 2015

مجلة الرابط؟

مجلة تقنية شهرية تهتم بمجالات البرمجة والتطوير بشكل خاص، وبالمعلوميات بشكل عام، وتتطرق بشكل ثانوي لقضايا الأدب والدين.

حقل الواقع

خالد السعداني

سيقولون لك: من أراد المعالي، فليسهر الليالي. أما أنا فأقول لك: من أراد المعالي فليشحذ الهمة، لأن اجتهاد ساعة بهمة عالية خير من سهر الليالي بهمة خائرة.

المحتويات

- افتتاحية الرابط: ليست رابطا بل انصهارا
- ثورة الخدمات: WCF نموذجا (الجزء الثاني)
- قصة ظهور قاصمة الظهور OOP (الجزء الثاني)
- الفرق بين Abstract Classes وبين Interfaces
- جولات في أعماق PHP (الفرق بين GET و POST)
- لست طبيبا أنا مريض مثلكم
- كيف برمجت أول لعبة أندرويد
- سبيل النجاح
- السلسلة التعليمية في التطور التكنولوجي
- التفكير البرمجي
- أدبيات فن تصميم المواقع
- أيقظها فإن عادت للنوم فنم
- قهوة الرابط
- كلمة ختامية

مجلة الرابط

ليست رابطا بل انصهارا

رئيس التحرير

الحمد لله ابتداء وانتهاء وبعد:

يقول الله عز وجل في محكم تنزيله: "إن يعلم الله في قلوبكم خيرا، يؤتكم خيرا". صدق الله رب العالمين.

لم تزل سنن الله عز وجل جارية في خلقه، ولسوف تزال جارية إلى أن يطوي الله الأرض كطي السجل، لقد رأينا من سعى في الغمرات كيف اصطلى بنارها، ورأينا من سعى في الخيرات كيف رزق من ثمارها.

فكان من جملة ما رأينا هذه المجلة التي بين يديك، فقد طرحناها أول ما طرحناها سعيا في الخير وطلبا للمنفعة العامة، فكان من جزيل نعم الله علينا أن بارك فيها وكتب لها النجاح من أول عدد.

أقبل الإخوة والأخوات على العدد الأول إقبالا شديدا ففاق عدد التحميلات ثلاثة آلاف تحميل في مدة تقل عن عشرين يوما، إضافة إلى الانطباعات الحسنة والآراء الطيبة حول المجلة ومحتواها وتصميمها.

المدهش في الأمر ليس هذا الإقبال الكبير، ولكن المدهش تحول المجلة من رابط بيننا وبين الجمهور إلى انصهار وذوبان، فصرت أجد متعة بالغة في الاشتغال عليها بغرض نفع وإسعاد الإخوة المتابعين، وكذلك هم فقد صاروا يكثفون الرسائل والانطباعات الإيجابية لتحفيزي على مزيد من العطاء، فأقولها اليوم بملء فِيَّ:

المجلة ليست رابطا، بل انصهارا.

انصهار لأنها محت الحدود بين القارىء والكاتب فصارت سعادة كل منهما من صميم سعادة الآخر..

انصهار لأنها لا تستهدف فئة من المبرمجين دون فئة، بل تستهدف الجميع باختلاف توجهاتهم ومستوياتهم..

انصهار لأنها تكلم الناس بلغتهم الأم حول التكنولوجيا، فمحت حدود الجغرافيا ووهم أن العربية لغة من في القبور..

انصهار لأنها قضت على كل كبر وعجب، وفتحت بابها لكل من يريد السعي في المنفعة..

الجديد في هذا العدد الثاني من مجلة الرابط، هو هبوب عدة إخوة أفاضل من سائر بقاع العالم الإسلامي إلى المشاركة والإسهام، فجازاهم الله خيرا أجمعين على جهودهم المباركة ومشاركاتهم الطيبة.

هذا ونسأل الله عز وجل أن يجعل هذا العمل خالصا لوجهه الكريم، وألا يجعل للنفس ولا للشيطان فيه حظا، إنه ولي ذلك والقادر عليه.

دام لكم البشر والفرح

تقنية WCF: ثورة الخدمات

(الجزء الثاني)

التعرف على عناصر WCF الأساسية

خالد السعداني

في الجزء الأول من سلسلة ثورة الخدمات تطرقنا إلى الأطوار التي مرت منها بنيات الأنظمة المعلوماتية، واستعرضنا معا بنية الأنظمة الموجهة خدماتيا أو خدميا Service Oriented Architecture المعروفة اختصارا ب SOA، مما استدرجنا إلى الحديث عن الأنظمة الموزعة Distributed Systems وما تحتاجه من تقنيات لتبادل البيانات بين مختلف مكوناتها المتباعدة جغرافيا.

وفي معرض حديثنا عن هذه التقنيات ركزنا على تقنية WCF، تعرفنا عليها، على تاريخها، وعلى هدفها الأساسي المتمثل في السماح بنقل الكائنات Objects عبر الشبكات المحلية للمؤسسات أو عبر الانترنت من أجل تمكين التطبيقات التي تحتاج هذه الكائنات من الوصول إلى وظائفها Methods واستعمالها.

وكخلاصة لكل ما سلف يمكننا القول أن تقنية WCF هي آلية أنتجتها ميكروسوفت من أجل السماح بإنشاء أنظمة ذات بنية موجهة خدميا SOA، والغرض منها هو تبادل البيانات بين مختلف مكونات النظام عبر الوسيط الشبكي (إما انترنت أو شبكة محلية).

مدار كل هذه التقنية على الخدمات Services، التي تمثل بدورها تطبيقا يستقبل طلبات العملاء Requests ويقوم بإرجاع الرسائل Responses إليهم، بصرف النظر عن التقنية المستخدمة على مستوى العميل، إذ من الممكن أن ننشئ تطبيقا بلغة سي شارب أو فيجوال بيسك، ثم يقوم تطبيق مبرمج بلغة جافا أو php بالاستفادة

منه، وهو ما يعرف عامة في خدمات الويب بالتوافقية Interoperability.

لكن كيف نقوم بإنشاء Service؟ كيف نستطيع استدعاءه؟ كيف نعرف عنوان الوصول إليه؟ كيف نعرف أسماء الوظائف التي يقدمها؟ ..

في الحقيقة الجواب على هذه الأسئلة ليس حكرا على تقنية WCF فقط، بل قد سبقتها تقنيات أخرى وقامت باعتماد نمط موحد تفهمه جميع التقنيات، فعلى سبيل المثال تقنية SOAP اتخذت من لغة XML وسيطا للتواصل بين مختلف التطبيقات المتعددة التقنيات.

أما في تقنية WCF فالتواصل بين التطبيقات يتم عبر ما يسمى بنقاط النهاية Endpoints، وعلى كل نقطة نهاية توفير المعلومات الثلاثة التالية المعروفة اختصارا ب ABC:

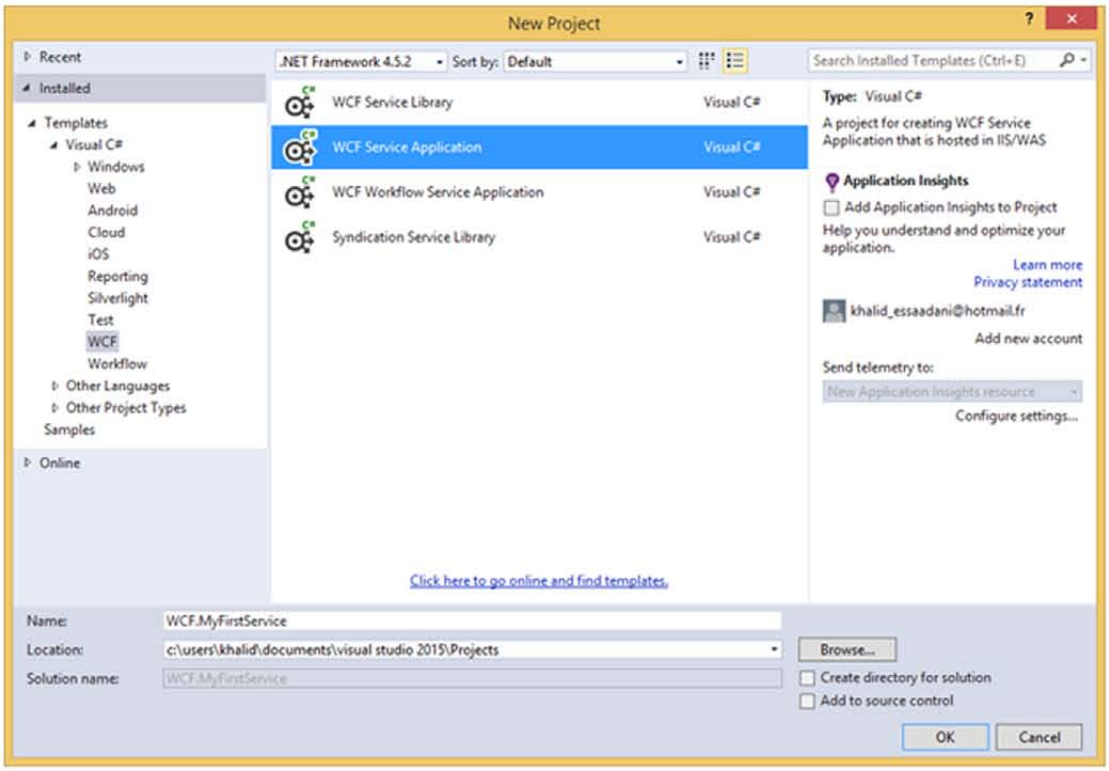
1. العنوان Address: من أجل معرفة رابط الوصول إلى Service
2. الربط Binding: من أجل تحديد خصائص Service، مثلا ماهو بروتوكول الاتصال المعتمد؟ ماهو مستوى الحماية؟ ماهو ترميز Service؟ وما إلى ذلك.
3. الاتفاق Contract: يشمل الاتفاق تعريفا لكافة مكونات Service، تعريف الوظائف، الخصائص... إذ يكفي قراءة الاتفاق لمعرفة ماهي عناصر Service الذي يقوم بتطبيقها.

من إيجابيات هذا التقسيم أنه يسمح بتوزيع المهام، بحيث يتكلف كل شخص حسب تخصصه بجزء من أجزاء الثلاثية أعلاه، مثلا إنشاء الاتفاق وتعريف الوظائف المكونة لل Service يبقى من اختصاص المطور Developer، بينما تحديد خصائص الربط Binding وعنوان الوصول إلى Service فتبقى ضمن مهام مدير الشبكة Network Administrator.

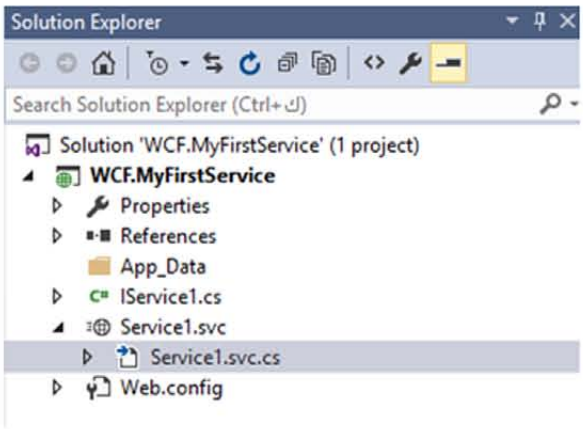
تسمح تقنية WCF بالتواصل مع Service عبر عدة قنوات للاتصال (http, TCP, ...) مما يعني أنه من الممكن فتح عدة نقاط نهاية Endpoints للاتصال ب Service وبالتالي يستطيع العميل اختيار طريقة التواصل مع Service التي تناسبه.

إنشاء Service باستعمال تقنية WCF:

لندخل إلى برنامج فيجوال ستوديو، ونقوم بإنشاء مشروع جديد من نوع WCF Service Application كما توضح الصورة التالية:



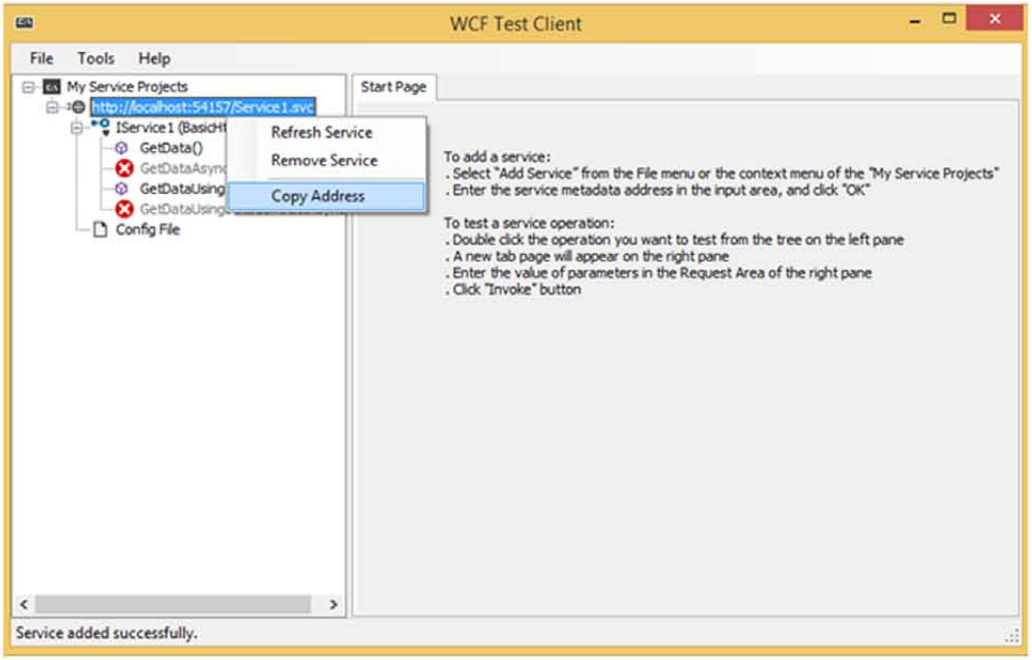
بعد أن يتم تهيئة بيئة التطوير، ستلاحظ أن المشروع قد جاء معه بعدة ملفات توضحها الصورة التالية:



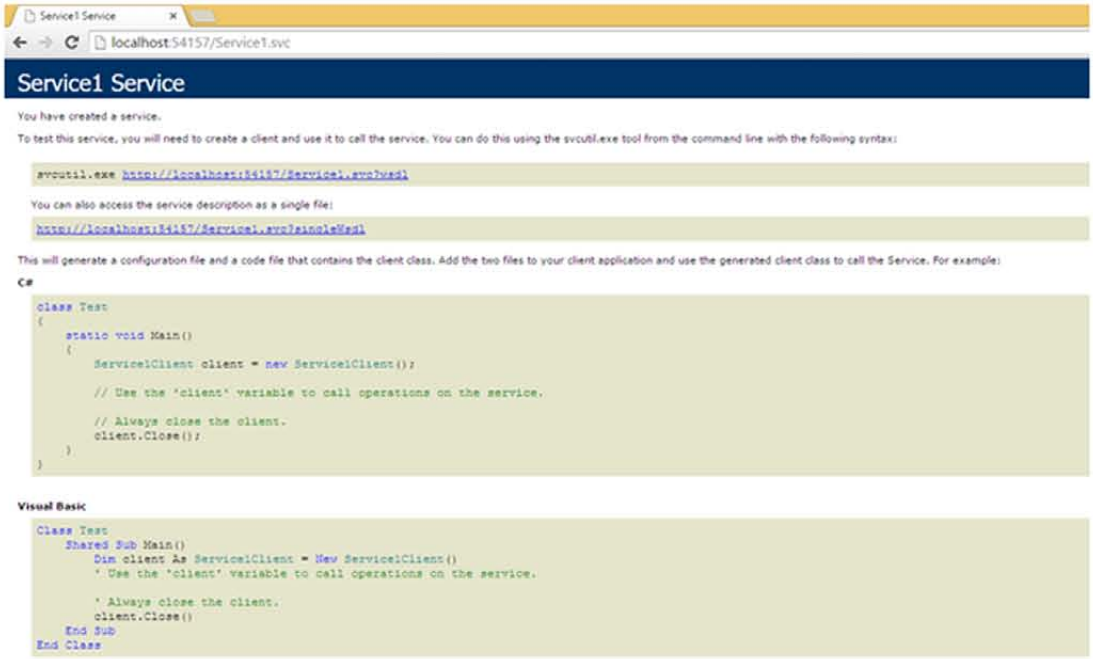
تحدثنا قبل قليل عن التقسيم ABC (Address, Binding, Contract) نظريا. الآن الصورة التي تراها أعلاه هي تجسيد لذلك. لكن هذا لن يمنعنا من شرح أهم الملفات أعلاه وهي كالآتي:

1. الملف IService1.cs: وهي أول مرحلة ينبغي أن نمر بها. بحيث ينبغي أن نقوم بتعريف جميع الوظائف التي سيقدمها Service على شكل واجهة Interface (لمن لا يعرف الواجهات يرجى قراءة مقالة "وجوه الائتلاف والاختلاف بين الفئات المجردة والواجهات" وهي منشورة في هذا العدد).
تسمى هذه الواجهة عادة ب" الاتفاق Contract". وهي عبارة عن ملف بامتداد .vb أو .cs حسب لغة البرمجة المستعملة. وكل الوظائف المعرفة على مستواها تكون مسبوقة بالواصفة Operation-Contract
2. الملف Service1.svc: يسمح لنا هذا الملف بنشر Service على رابط معين URL.
3. الملف Service1.svc.cs: هذا الملف عبارة عن فئة Class تقوم بتطبيق الواجهة IService وبالتالي يلومها إعادة تعريف جميع وظائفها. يمكننا تبسيط المعنى أكثر فنقول أن الواجهة تقوم بوصف الوظائف Specification. بينما تقوم الفئة التي تعمل implement لهذه الواجهة بتزويد الوظائف المعلن عنها بالأوامر البرمجية اللازمة وهو ما يعرف ب Implementation.
4. الملف Web.config: يسمح لنا هذا الملف بالتحكم في إعدادات Service ونقاط النهاية end-points.

إذا قمنا بتنفيذ التطبيق عبر الضغط على الزر F5. ستلاحظ أن التطبيق فتح على تطبيق اسمه WCF Test Client. يمكننا نسخ رابط Service عبر الضغط بيمين الفأرة على الرابط كما يلي:



ثم نذهب إلى أحد متصفحات الويب (غوغل كروم مثلا) ونقوم بلصق الرابط لتظهر لنا الصفحة التالية:



كما تلاحظ فالصورة أعلاه تخبرنا أنه تم إنشاء Service وأنه جاهز للاستخدام من قبل العميل. كما تقدم لنا نماذج لكيفية استخدام Service على مستوى العميل باستخدام لغتي الفيجوال بيسك و سي شارب.

الآن لنعد إلى المشروع، ونقوم بالدخول إلى الواجهة IService1.cs:

```
[ServiceContract]
public interface IService1
{
    [OperationContract]
    string GetData(int value);

    [OperationContract]
    CompositeType GetDataUsingDataContract(CompositeType composite);
}
```

المفترض أول شيء يقع نظرنا عليه هو الواصفة [ServiceContract] التي تسبق عملية التصريح عن الواجهة وتدل على كون هذه الأخيرة تمثل اتفاقا Contract لخدمات WCF Service. وكذلك الواصفة [OperationContract] التي تسبق التصريح عن الوظائف وتدل على أن هذه الأخيرة جزء من الواجهة التي تمثل اتفاقية الخدمة Service Contract.

تحتوي هذه الواجهة التي تم إنشاؤها افتراضيا في المشروع على وظيفتين اثنتين. الأولى اسمها Get-Data وتستقبل برامترا من نوع رقمي اسمه value. والثانية اسمها GetDataUsingDataContract وهي تستقبل برامترا من نوع CompositeType اسمه composite.

نكتفي بهذا القدر خلال هذا الجزء ونلتقي في الفصل المقبل إن شاء الله من العدد الثالث من مجلة الرابط لنتعرف حينها على كيفية إنشاء تطبيق من نوع WCF والاستفادة منه.

دام لكم البشر والفرح والسلام عليكم ورحمة الله وبركاته.

يتبع

فائدة

أيها الاعزاء لقد أصبحت البرمجة في وقتنا الحاضر أداة ذللت بها الكثير من الصعاب التي نواجهها في حياتنا العملية فوفرت لنا الكثير من الجهد والمال من خلال تطبيقات ومواقع ويب صممت لتنجز اعمالاً بكل يسر وسهولة بعيدا عن الطرق التقليدية التي استنزفت جهد ومال من كان يتعامل بها، هذه التطبيقات ومواقع الويب من قام بصنعها هم أناس من بني البشر يعتبرهم البعض خارقين ويمتلكون عقولا تختلف عن عقل الانسان العادي وفي الغالب هم ليسوا كذلك بل هم أناس عاديون وضعوا صوب اعينهم هدفا فسعوا بكل ما اوتوا من قوة لتحقيقه فكان لهم ما ارادوا, **انهم المبرمجون.**

بقلم / إدريس الزبيدي

تقرير مفصل

قصة ظهور قاصمة الظهور

(البرمجة الكائنية التوجه)

الجزء الثاني

الجزء الثاني

قصة ظهور قاصمة الظهور

البرمجة الكائنية التوجه

Object Oriented Programming

خالد السعداني

رأينا في الجزء الأول من السلسلة المكتوبة "قصة ظهور قاصمة الظهور" نبذة موجزة عن تاريخ البرمجة الكائنية التوجه Object Oriented Programming، وتركنا الخوض في تفاصيلها إلى الأجزاء المقبلة والتي يمثل هذا العدد أحد عارضي هذه الأجزاء، إذ سنتناول فيه بحول الله وقوته بعض المفاهيم الأساسية للبرمجة الكائنية التوجه سعيا منا إلى التدرج في عرض كل مفاهيمها إلى أن نأتي عليها.

أما لغة البرمجة التي سنعتمدها في شرح هذه المفاهيم فهي لغة الفيجوال بيسك دوت نيت، علما أنه يمكن إسقاط هذه الشروحات على باقي اللغات التي تتبنى نمط البرمجة الكائنية التوجه.

في نسخة الدوت نيت من الفيجوال بيسك التي ظهرت في بدايات عام 2002 تم إقحام نمط البرمجة كائنية التوجه كنمط برمجي أساسي في البنية الجديدة لهذه اللغة، الشيء الذي أدى إلى إضافة مجموعة من المفاهيم الجديدة المرتبطة بهذا النمط البرمجي مثل الفئات Classes والكائنات Objects والمشيدات Constructors والمهدمات Destructors والخصائص Properties والوظائف Methods والتغليف Encapsulation والوراثة Inheritance و تعدد الأشكال Polymorphism، وإعادة التعريف Overriding، والتظليل Shadowing وغير ذلك من المفاهيم المتعلقة بالبرمجة الشيئية.

كثرة المفاهيم الكائنية وسهولة الخلط بين مسمياتها مثل Overloads و Overrides و Overridable يتسبب في جعلها عصية على الفهم، وإن تم استيعابها تكون عصية على التذكر، لذلك سنحاول في غضون هذه السلسلة المكتوبة التطرق إلى كل مفهوم بأسلوب سلس.

ذكرنا فيما مضى أن البرمجة الكائنية التوجه Object Oriented Programming هي أسلوب برمجي يخول لنا كتابة الشفرات على شكل فئات Classes وكائنات Objects. ليصبح الكود شبيها بالحياة الواقعية حيث يصبح النوع الشامل عبارة عن فئة Class وكل نسخة من هذا النوع تصبح عبارة عن Object. بينما الدوال والإجراءات تصبح عبارة عن وظائف Methods لهذه الكائنات.

تعالوا بنا نبسط هذه الكلمات. الفئة Class عبارة عن تمثيل شامل لنوع معين. فلو أخذنا على سبيل المثال "سيارة Car" فهي تمثيل شامل لنوع معين. لكن حينما نحدد نوعا معينا كأن نقول "ميرسيدس" فنحن هنا نتكلم عن كائن Object مستنسخ من الفئة Car.

لترسيخ الفكرة في أذهاننا فكر في أي تمثيل عام لكائن معين. خذ "الحاسوب" مثلا. أليس الحاسوب نمطا عاما لمجموعة من الأجهزة التي تشترك في نفس الخصائص والوظائف؟ أليست كل الأجهزة التي نسميها حواسيبا لها نوع وشكل معين؟ أليس من وظائف هذه الأجهزة "الاشتغال والتوقف وتخزين ومعالجة البيانات"؟ إذن يمكننا أن نعتبر الحاسوب فئة. لأنه يقدم نوعا قائما بذاته لمجموعة من الكائنات التي تؤدي نفس الدور وتكتسي نفس الخصائص. بنفس القياس لو حددنا حاسوبا بعينه: خذ مثلا " Compaq " فنحن هنا خرجنا من التعميم ودخلنا في التخصيص إذن نحن بدأنا نتكلم عن الكائنات Objects.

في لغة الفيجوال بيسك دوت نيت. نستطيع إنشاء فئة عبر كتابة الكلمة المحجوزة Class متبوعة باسم الفئة. فلو أردنا إسقاط الكلمات السابقة تطبيقيا فإن كود إنشاء الفئة "حاسوب CLS_COMPUTER" كما يلي:

```
Class CLS_COMPUTER

End Class
```

كل فئة قد تتكون من ثلاثة أجزاء:

الخصائص Properties: وهي مواصفات الفئة وخصائصها ومميزاتها

الوظائف Methods: الأعمال والسلوكيات التي تقوم بها الفئة

الأحداث Events: أفعال مرتبطة بالكائنات تؤدي عند حصولها إلى استدعاء وظائف معينة.

الخصائص Properties:

وهي ما تتميز به الفئة من خصائص ومميزات. على سبيل المثال حينما نتحدث عن الفئة "حاسوب Computer" فيمكننا اعتبار لون الحاسوب. ونوعه. وتاريخ تصنيعه. وثمن بيعه من الخصائص المميزة لهذه الفئة. لذلك يمكننا كتابتها داخل الفئة على شكل متغيرات تسمى في عرف البرمجة الكائنية بالحقول Fields أو Attributes.

في المثال الآتي قمنا بكتابة الحقول المميزة لفئة الحاسوب:

```
Class CLS_COMPUTER
    'Fields / Attributes
    Public Model As String
    Public Price As Decimal
    Public Date_of_Manufacture As DateTime
End Class
```

لاحظ أننا قبل كتابة أسماء حقول الفئة، وضعنا الكلمة Public التي تنتمي إلى أسرة محددات الوصول Access Modifiers، وسميت كذلك لأنها تمكننا من تعريف مجال الوصول إلى هذا الحقل.
في لغة الفيجوال بيسك دوت نيت تتكون أسرة محددات الوصول من أربعة كلمات وهي كما يلي:

محدد الوصول شرحه

Private وتعني أن الحقل سيبقى معروفا فقط على مستوى الفئة التي تحتويه، ولو أردت استعماله خارج مجال الفئة فلن يمكنك ذلك لأن محدد الوصول إليه خاص Private.

Public إذا أعلنا عن متغير أو وظيفة أو أي عنصر من عناصر الفئة بالكلمة Public، فهذا يعني أنه يمكننا الوصول إليه داخل الفئة التي تحتويه وخارجها كذلك، بل وخارج المشروع الذي توجد فيه هذه الفئة أيضا، بمعنى لو لدي مجموعة من المشاريع وواحد منها يوجد به عنصر معلن عنه بمحدد الوصول Public فيعني ذلك أنه يمكنني استخدامه داخل كافة المشاريع الأخرى.

Friend مثلها مثل Public تسمح لنا بالوصول إلى العنصر من داخل وخارج الفئة التي تحتويه، لكن مجال الوصول يتوقف عند المشروع الذي يضم هذه الفئة، بحيث لو أردت استخدام هذا العنصر داخل مشروع آخر فلن يمكنني ذلك على خلاف محدد الوصول Public.

Protected هذه الكلمة تعني أن العنصر المعلن عنه بها يمكننا الوصول إليه فقط على مستوى الفئة التي تحتويه وكذلك الفئات المشتقة منها (سنتطرق إلى مفهوم الاشتقاق / الوراثة في فصل قادم)

لاستنساخ كائن من هذه الفئة يكفي أن نعلن عنه بنفس الطريقة التي نعلن بها عن المتغيرات مع استبدال نوع البيانات باسم الفئة مع ضرورة كتابة الكلمة New كما يلي:

```
'Instanciate new object from Computer Class
Dim Computer As New CLS_COMPUTER
```

بعد ذلك يمكننا إسناد القيم لحقول الفئة أو استعراضها كما يلي:

```
'Instanciate new object from Computer Class
Dim Computer As New CLS_COMPUTER

'إسناد القيم إلى الحقول
Computer.Model = "Acer"
Computer.Price = 345.5
Computer.Date_of_Manufacture = New DateTime(2006, 12, 24)

'الحصول على قيم الحقول
MessageBox.Show(String.Format("{0},{1},{2}",
                Computer.Model,
                Computer.Price,
                Computer.Date_of_Manufacture))
```

يوجد في البرمجة الكائنية مفهوم مهم يسمى Encapsulation والذي نستطيع الاصطلاح عليه بالتغليف. ويعني هذا المفهوم أن نخفي الكود الخاص بالفئة عند التعامل مع كائنات مستنسخة منها. فبدل أن يستعرض المبرمج الثاني أكواد فئاتنا يصبح قادرا فقط على مشاهدة أسماء الوظائف والخصائص فقط. لذلك ينبغي أن نعلن دائما عن حقول الفئة بمحدد الوصول Private لكي لا يراه المبرمج الثاني، ونستعيض عن اسم الحقل بمفهوم جديد يسمى الخصائص Properties تكون عامة الوصول Public بينما تصبح الحقول خاصة الوصول Private.

كل حقل من حقول الفئة علينا إنشاء خاصية له حسب دوره. فإن كان يعطي معلومة فقط أعلنا عن الخاصية من نوع ReadOnly فقط. أما إن كان يستقبل القيمة فقط أعلنا عن الخاصية من نوع WriteOnly. وإلا فإننا في الوضع العادي نستغني عن الكلمتين معا. ونعلن عن الخاصية بشطريها.

كل خاصية مكونة من جزءين. جزء يمكننا من الحصول على القيمة المحفوظة في الحقل ويسمى Getter. وجزء يمكننا من إسناد قيمة ما لهذا الحقل ويسمى Setter.

لإنشاء خاصية لحقل ما. ينبغي أن نكتب محدد الوصول أولا (والأفضل أن يكون Public احتراما لدور الخصائص وهو تغليف الحقول) ثم الكلمة المحجوزة Property متبوعة باسم الخاصية وبنوعها (نفس نوع الحقل المراد إنشاء الخاصية له). وهذا مثال على إنشاء خصائص لكافة حقول الفئة "حاسوب CLS_COMPUTER"

```
Class CLS_COMPUTER
    'Fields / Attributes
    Private Model As String
    Private Price As Decimal
    Private Date_of_Manufacture As Date
    'Properties الخصائص
    Public Property Model_Property As String
        Get
            Return Model
        End Get
        Set(value As String)
            Model = value
        End Set
    End Property
    Public Property Price_Property As Decimal
        Get
            Return Price
        End Get
        Set(value As Decimal)
            Price = value
        End Set
    End Property
    Public Property Date_of_Manufacture_Property As Date
        Get
            Return Date_of_Manufacture
        End Get
        Set(value As Date)
            Date_of_Manufacture = value
        End Set
    End Property
End Class
```

لو دققت جيدا في الكود أعلاه، ستلاحظ أن الخصائص معلن عنها بمحدد الوصول Public، بينما الحقول أصبحت Private، وهذا وجه من وجوه التغليف Encapsulation بحيث يتم حماية بيانات الكائن داخل الفئة ولا يستطيع المبرمج الثاني التعامل معها إلا من خلال الخصائص والوظائف.
وستلاحظ كذلك أن لكل خاصية نوع بيانات من نفس نوع الحقل الذي تمثله، وأن كل خاصية مكونة من جزءين، الجزء الأول عبارة عن Getter يمكننا من الحصول على قيمة الحقل، والجزء الثاني عبارة عن Setter يمكننا من إسناد القيمة إليه.
تمكننا الخصائص من قراءة و إسناد القيم لحقول الفئة، بحيث يمكننا التعامل مع الكائن المستنسخ من الفئة "حاسوب CLS_COMPUTER" كما يلي:

```
'Instanciate new object from Computer Class
Dim Computer As New CLS_COMPUTER

'Setting Values
Computer.Model_Property = "Acer"
Computer.Price_Property = 345.5
Computer.Date_of_Manufacture_Property = New DateTime(2006, 12, 24)

'Getting Values
MessageBox.Show(String.Format("{0},{1},{2}",
Computer.Model_Property,
Computer.Price_Property,
Computer.Date_of_Manufacture_Property))
```

يمكننا فصل الجزءين المكونين للخاصية عن بعضهما البعض، أو الاكتفاء بأحدهما حسب طبيعة الحقل، فلو أردنا أن يكون أحد الحقول للقراءة فقط نستغني عن الجزء Set ونعلن عن الخاصية بالكلمة المحجوزة ReadOnly كما يلي:

```
Public ReadOnly Property GetModel As String
    Get
        Return Model
    End Get
End Property
```

يتبع...

مغارات نخشى وطأها

أوجه الاختلاف والائتلاف بين الفئات المجردة Abstract Classes والواجهات Interfaces

خالد السعداني

الفئات المجردة Abstract Classes هي تمثيل مجرد لنوع معين، ومن مقتضيات هذا التمثيل ألا يكون له وجود واقعيا إذ يبقى في حيز التجريد والاصطلاح فقط، مما يترتب على هذا المعنى من إلغاء لعملية الاستنساخ Instanciation، إذ من المعلوم أن الفئات المجردة في البرمجة الكائنية التوجه لا تسمح باستنساخ كائنات منها ويبقى دورها محصورا في عملية تعريف العناصر الأساسية للفئات بغرض توريثها إلى الفئات المشتقة.

ويمكن للفئة المجردة أن تشتمل على وظائف مجردة abstract methods قابلة لإعادة التعريف على مستوى الفئات المشتقة، كما يمكنها أيضا ألا تشتمل على أية وظيفة مجردة، والوظائف المجردة هي وظائف لا تحتوي على أية أوامر برمجية Imple-mentation، بل تعرض فقط تعريف الوظيفة أو ما يعرف بالتوقيع Signature كما يوضح المثال التالي:

```
abstract class Person
{
    string _name;
    public Person (string name)
    {
        _name = name;
    }
    abstract public void Work();
}
```

الفئة Person فئة مجردة وعلامة ذلك أمر التعريف abstract الذي تقدمها، وهي تحتوي على حقل من نوع نصي اسمه _name يأخذ قيمته في مشيد الفئة Constructor (علما أن المشيد في الفئات المجردة لا معنى له، بيد أن دور المشيد هو تشييد الكائنات في زمن الاستنساخ، ومادام الاستنساخ غير ممكن من الفئات المجردة فلا قيمة لوضع المشيد هنا، لكن يتم وضعه لتحديد قيم بدئية للحقول المشتركة بين جميع الفئات المشتقة التي ترث من الفئة المجردة، ويمكن تغيير قيمة الحقل في المشيدات الفرعية عبر الكلمة base كما سيأتي معنا).

بعد ذلك قمنا بتعريف وظيفة مجردة abstract method أسميناها Work، هاته الوظيفة سنقوم بإعادة تعريفها في الفئات المشتقة عبر استخدام الأمر Override كما يعرض الكود التالي:

```
class Programmer:Person
{
    string _speciality;

    public Programmer(string speciality, string name) : base(name)
    {
        _speciality = speciality;
    }

    public override void Work()
    {
        Console.WriteLine("I am Programmer :) ");
    }
}
```

في الكود أعلاه، قمنا بإنشاء فئة اسمها Programmer مشتقة من الفئة الرئيسية المجردة Person. داخل هاته الفئة المشتقة قمنا بإسناد قيمة بدئية للحقل _name من جديد عبر الأمر base الذي يحيل إلى الفئة الأصلية، كما قمنا بإعطاء قيمة بدئية لحقل جديد أسميناه Speciality.

ثم في الجزء الأهم، قمنا بإعادة تعريف الوظيفة المجردة Work من خلال الأمر Override، وبالتالي عند استدعاء هاته الوظيفة عبر كائن مستنسخ من الفئة Programmer سيتم تنفيذ الكود الموافق لها، كما يوضح الكود التالي:

```
class TestProg
{
    static void main()
    {
        //Create new instance from Programmer Class
        Programmer programmer = new Programmer("Desktop", "Khalid ESSAADANI");

        //Call Work method from derived class
        programmer.Work(); //Print: I am Programmer
    }
}
```

إذن كخلاصة، فالفئات المجردة هي عبارة عن فئات تقوم بتعريف العناصر العامة لنوع معين، ولا تسمح بعملية الاستنساخ المباشر منها بل تسمح بذلك فقط عبر آلية التوريث / الاشتقاق In-heritance بحيث يمكننا استنساخ كائنات فقط من الفئات الفرعية المشتقة من هذه الفئات المجردة.

الواجهات Interfaces:

تقوم الواجهة Interface بتعريف الوظائف التي على الفئات - التي ستقوم بتطبيقها – إعادة تعريفها، وكما هو الحال في الوظائف المجردة فإن الواجهات لا تحتوي على أية أوامر برمجية Implemen-tation، بل تحتوي فقط على تعريف لتوقيعات الوظائف الواجب إعادة تعريفها على مستوى الفئات التي ستستخدم هاته الواجهة.

ولعل سؤالا ملحا ينخر في ذهنك الآن: إذن لماذا تصلح الواجهات مادامت تؤدي تقريبا نفس الدور الذي تؤديه الفئات المجردة؟

سؤال وجيه و حق له أن يطرح، وكجواب مبدئي يمكننا القول، أن الواجهات في اللغات الحديثة مثل سي شارب وجافا وفيجوال بيسك دوت نيت، أتت كحل حاسم لأزمة الوراثة المتعددة Multiple inheri-tance، إذ من المعلوم أن هذه اللغات الثلاثة التي ذكرناها لا تسمح بالوراثة المتعددة، بمعنى أن فئة

معينة لا تستطيع أن ترث من أكثر من فئة رئيسية، بيد أن التسلسل المسموح به هو فئة فرعية ترث من فئة رئيسية واحدة.

فلما كان الأمر كذلك، قامت هذه اللغات بتقديم بديل للوراثة المتعددة متمثلا في الواجهات، حيث صار للفئة الواحدة أن تطبق واجهة أو عدة واجهات في الوقت عينه وبالتالي يكفي أن ننشئ الواجهات التي سنحتاجها ونضع فيها ما نشاء من تعريفات للوظائف Specification، شريطة ألا يتعدى التعريف حدود التوقيع Signature لأن الواجهات لا تحتوي على أوامر برمجية Implementation، ثم بعد ذلك نأتي إلى الفئة المستهدفة ونعيد تعريف عناصر الواجهات عليها بكيفية قريبة جدا من طريقة الوراثة، حيث في لغة الفيجوال بيسك دوت نيت نقوم بتطبيق الواجهة في فئة ما من خلال الكلمة المحجوزة Implements وكذلك في لغة جافا تتم عملية التطبيق عبر استخدام الكلمة المحجوزة implements، بينما في لغة سي شارب، يبقى رمز تطبيق الواجهات هو نفسه رمز الوراثة (نقطتان هكذا :).

وهذا مثال على إنشاء واجهة في لغة سي شارب:

```
public interface IVehicle
{
    string Model { get; set; }
    DateTime ManuFacturedDate { get; set; }
    byte Age();
}
```

الواجهة أعلاه اسمها IVehicle / المركبات، تحتوي على تعريف لخاصيتين وهما موديل المركبة Model و تاريخ التصنيع ManuFacturedDate، وعلى وظيفة اسمها Age، لو لاحظت معي فإننا اكتفينا في تعريف الخصائص بذكر get و set فقط دون تحديد التفاصيل، لأن الواجهات كما ذكرنا تشمل التعريف Specification وليس الكود Implementation.

إذا أردنا تطبيق هاته الواجهة في فئة ما، فالمسألة بسيطة وصيغتها كما تقدم شبيهة بصيغة الوراثة أي كما يلي:

```
class Car : IVehicle
{
    public string Model { get; set; }
    public DateTime ManuFacturedDate { get; set; }


    public byte Age()
    {
        return (byte)(DateTime.Now.Year - ManuFacturedDate.Year);
    }
}
```

جدير بالذكر أن تطبيق واجهة ما في إحدى الفئات يستلزم إعادة تعريف جميع العناصر التي قامت الواجهة بتعريفها، كما هو الحال في المثال أعلاه، فقد قمنا بتعريف الخصائص Model و ManuFac-turedDate وكذلك الوظيفة Age التي تسمح لنا بالحصول على عمر السيارة من خلال طرح تاريخ التصنيع من التاريخ الحالي سنويا.

ويمكن للفئة الواحدة أن تطبق عدة واجهات كما يعرض المثال الآتي:

```
public interface IVehicle
{
    string Model { get; set; }
    DateTime ManuFacturedDate { get; set; }
    byte Age();
}

public interface IColor
{
    string Color { get; set; }

    bool CompareColors(string color);
}
```

```
class Car : IVehicle, IColor
{
    //IVehicle
    public string Model { get; set; }
    public DateTime ManuFacturedDate { get; set; }


    public byte Age()
    {
        return (byte)(DateTime.Now.Year - ManuFacturedDate.Year);
    }


    //IColor
    public string Color { get; set; }

    public bool CompareColors(string color)
    {
        if (Color.Equals(color))
            return true;
        else return false;
    }
}
```

في المثال أعلاه، قامت الفئة Car بتطبيق واجهتين هما IVehicle و IColor، وأعادت تعريف جميع عناصرهما.

كخلاصة، الواجهات تقوم بتعريف وظائف وخصائص الفئات فقط دون كتابة الأوامر البرمجية، والغرض من استخدام الواجهات أساسا هو تعويض مفهوم الوراثة المتعددة، إذ لا يمكن للفئة الواحدة الوراثة من أكثر من فئة رئيسية، بينما يمكنها تطبيق أكثر من واجهة.

كما تسمح الواجهات بتوصيف شامل لجميع عناصر الفئات التي تستخدمها، إذ تحتوي على تعريف جميع الوظائف والخصائص.

الفروق بين الفئات المجردة والواجهات

1. الفئات المجردة يمكنها أن تحتوي على وظائف بأوامر برمجية Implementation، بينما الواجهات لا تسمح بذلك وتكتفي فقط بتقديم تعريفات للعناصر.
2. يمكن للفئات المجردة أن تحتوي على ما تحتويه الفئات العادية كالمشيدات Constructors والمهدمات Destructors بينما بنية الواجهات لا تسمح بذلك.
3. يمكن للفئة الواحدة أن ترث من فئة مجردة واحدة، بينما يمكنها أن تطبق أكثر من واجهة.
4. الفئات المجردة تبدأ بالكلمة abstract بينما تبدأ الواجهات بالكلمة interface.
5. على الفئة التي تستخدم واجهة ما، إعادة تعريف جميع عناصرها، بينما في حال الوراثة من فئة مجردة فهي ليست ملزمة بذلك.
6. يستحسن استخدام الفئات المجردة حينما نحتاج إلى إعادة تعريف بعض عناصر الفئة الرئيسية فقط على مستوى الفئات البنات، بينما يستحسن استخدام الواجهات حينما نحتاج إلى إعادة تعريف جميع العناصر على مستوى الفئة.

جولات في أعماق بحر الـ PHP

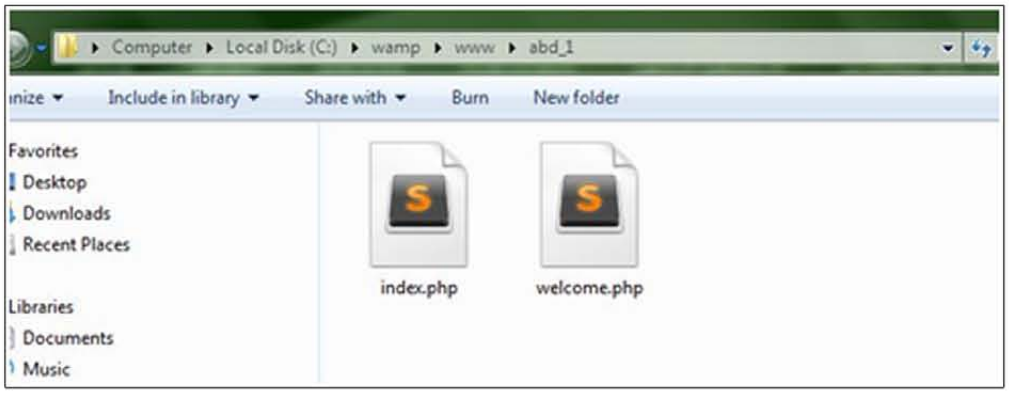
الدالتان GET & POST
أماكن استخدامهما
وخصائص كل منهما

عبد الغني جمعة - سوريا (حلب)

لا يخلو موقع ويب ديناميكي مبرمج بلغة php من هاتين الدالتين.فما فائدة استخدام هاتين الدالتين ومتى وأين يتم استخدام كل منهما؟ وهل يغني استخدام إحداهن عن الأخرى؟ هذا ما سنوضحه بحول الله مع بعض الأمثلة في هذه المقالة .

تستخدم الدالتين GET & POST لتمرير البيانات من صفحة إلى أخرى ضمن الموقع. وذلك عن طريق استقبال البيانات عبر المتغيرات وسأوضح ذلك من خلال مثال عملي عن آلية إحضار البيانات وتمريرها والتحكم بها من خلال الدالتين السابقتين..

لنفترض أنه لدينا صفحتان من نوع php ولتكن الأولى اسمها index.php والثانية welcome.php..



الصفحة الأولى تحتوي على form لإدخال البيانات ضمن حقلي الاسم وكلمة السر...
حيث تم تحديد اسم الصفحة التي ستستقبل البيانات من خلال القيمة المخزنة في المتغير action و سيتم إسناد قيمة الحقل التي سيدخلها المستخدم إلى المتغير name في كل حقل. وقد تم تحديد اسم الدالة التي ستقوم بإرسال البيانات المدخلة إلى الصفحة welcome.php والتي هي GET..

```
C:\wamp\www\abd_1\index.php • (abd_1) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
abd_1
index.php
welcome.php
welcome.php   index.php

1
2   <html>
3   <head>
4       <title>Test</title>
5   </head>
6   <body>
7
8       <form action="welcome.php" method="GET">
9           <input type="text" name="name" />
10          <input type="password" name="password" />
11          <input type="submit" name="send">
12      </form>
13
14  </body>
15  </html>
```

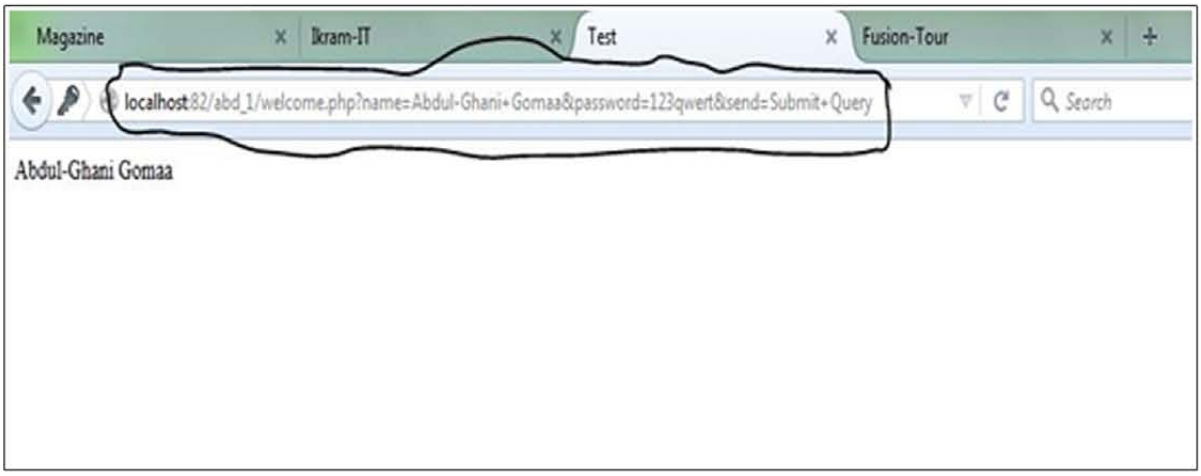
أما في الصفحة الثانية welcome.php سنقوم ببعض العمليات على البيانات التي تم جلبها من الصفحة index.php. كما هو موضح في الصورة حيث قمنا بعرض القيمة التي تم استقبالها عن طريق الدالة echo:

```
C:\wamp\www\abd_1\welcome.php • (abd_1) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
abd_1
index.php
welcome.php
welcome.php   index.php

1
2   <html>
3   <head>
4   <title>Test</title>
5   </head>
6   <body>
7       <?php
8
9       echo $_GET['name'];
10
11      ?>
12
13  </body>
14  </html>
```

ملاحظة: قد تم استخدام نفس الكود السابق في كلا الملفين، ولكن مع مراعاة تغيير الدالة من GET إلى POST لتوضيح الفرق بينهما.

الآن و من خلال تنفيذ الكود السابق بحالتيه كما ذكرنا سنوضح خصائص وآلية كل منهما.
حيث تم إنشاء هاتين الدالتين لتعملان وفق منهجيتين مختلفتين ولا يمكن القول أنه يمكن الاستغناء بإحداهن عن الأخرى لأن لكل واحدة أهميتها ..ولايمكن أبدا الاستغناء عن إحداهن عند كتابة الكود البرمجي في لغة php .
تستخدم الدالة GET لتمرير قيم الحقول المدخلة من قبل المستخدم إلى الصفحة الهدف. بطريقة يمكن من خلالها معرفة قيمة الحقل التي تم إرسالها بواسطة هذه الدالة. والتي ستظهر في شريط العنوان للمتصفح بعد الرابط كما هو موضح في الصورة:



أما عند استخدام الدالةPOST فسيتم إرسال البيانات إلى الصفحة الهدف ولكن دون إظهار أية معلومة عن تلك البيانات في شريط العنوان للمتصفح. كما توضح الصورة التالية:

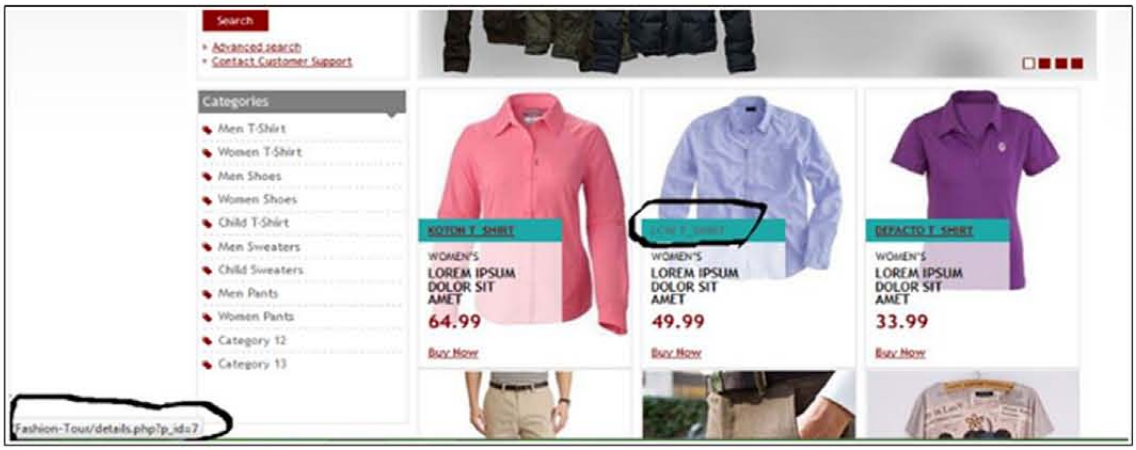
قد يقول البعض..مافائدة استخدام الدالةGET..والتي ستكشف البيانات للجميع ...سأقول لكم أن هذه الدالة يتم استخدامها فقط في الأماكن التي يفضل أن يتم إظهار البيانات فيها للمبرمج والمستخدم ...حتى يعرف ماهي القيم التي تم جلبها وماهي الصفحة الهدف المراد الذهاب إليها والتي ستكون مرهونة بقيمة المتغير المرسل بالدالة GET ...وهي ميزة هامة..وإظهار البيانات في هذه الحالة ..لن يؤثر كثيرا على ال security...أما في الأماكن التي تستدعي ال security فمن الضروري استخدام الدالةPOST.

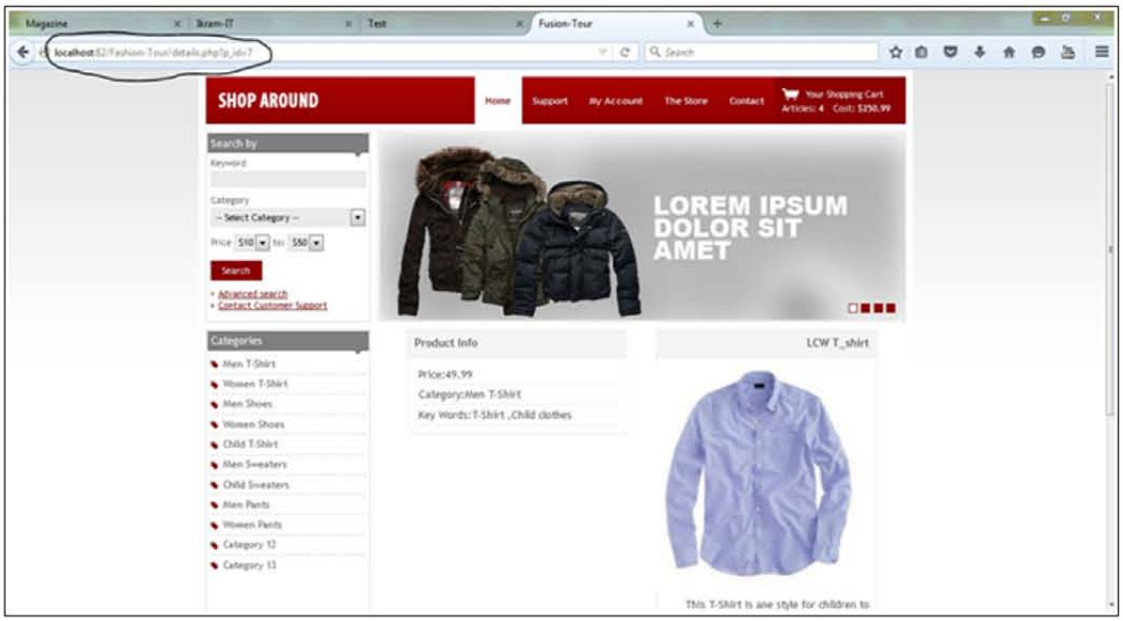
كل ما ذكر أعلاه عبارة عن توضيح عام لآلية العمل وكيفية ظهور النتائج.

إذاً كما شاهدنا أن هناك أهمية لاستخدام الدالتين كلٌّ في المكان والزمان المناسب. ويمكن تلخيص أماكن استخدامها بالمواضع التالية"سيتم نقاش المواضع من خلال مثال لموقع تجاري يتعامل مع عمليات الدفع الإلكتروني":

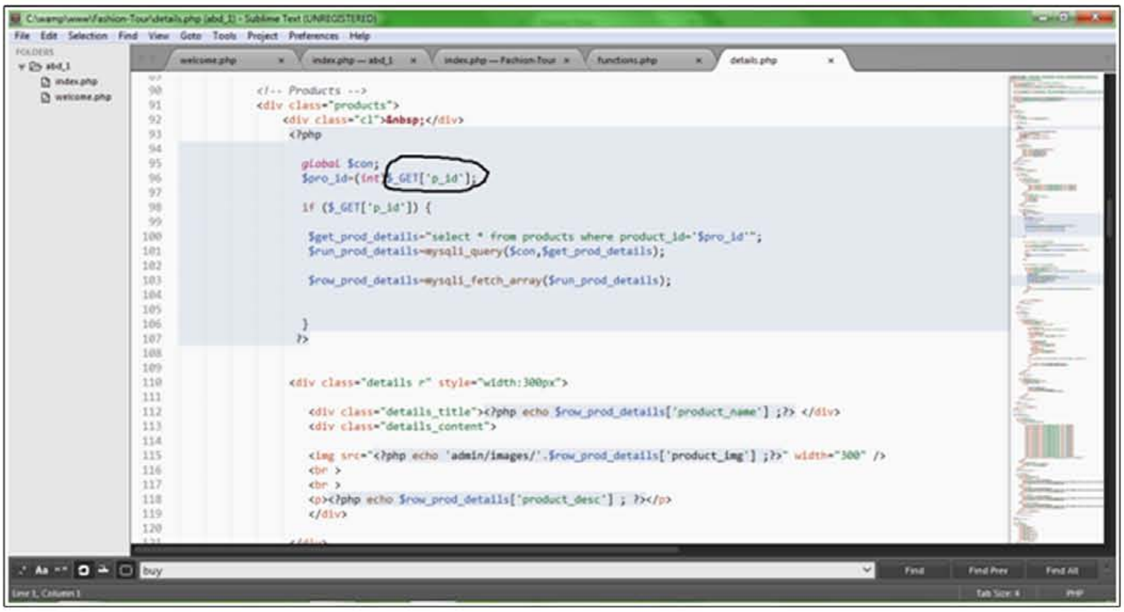
أولاً GET:

1. مثلا عند الضغط على منتج معين لمعرفة تفاصيل هذا المنتج سيتم إرفاق الID الخاص بهذا المنتج مع الرابط ليقوم المتصفح بالبحث عن المنتج المحدد في قاعدة البيانات. جلب تفاصيله وعرضها للمستخدم.

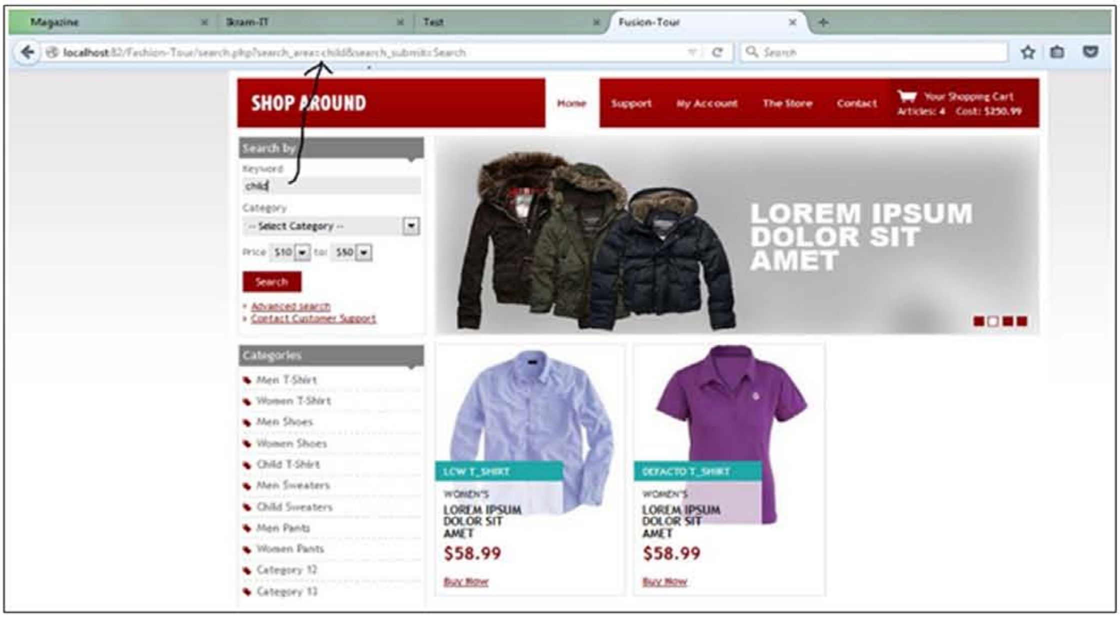




والكود هو كالتالي :



2. في عمليات البحث عن منتج معين ضمن قاعدة البيانات يفضل استخدام الدالةGET حيث أيضا يتم إقحام القيمة المراد البحث عنها ليتم استخدامها في استعلام البحث وإظهار النتائج الموافقة. حيث ستلاحظ في الصورة التالية أن القيمة التي تم إدخالها إلى form الإدخال..يتم عرضها في شريط العنوان.ويتم عرض النتائج في تلك الصفحة بعد تنفيذ الاستعلام:



والكود على الشكل التالي:

```
function get_product_by_search(){
    global $con;
    if (isset($_GET['search_submit'])) {
        $search_area=$_GET['search_area'];
        $get_prod_search="select * from products where product_keywords like'%$search_area%'";

        $run_prod_search=mysqli_query($con,$get_prod_search);

        if(mysqli_num_rows($run_prod_search)>0){
        while ($row_prod_search=mysqli_fetch_array($run_prod_search)) {

                echo '
                        <li>
                            <a href="#"><img src="admin/images/'.$row_prod_search['product_img'].'" alt="" /></a>
                            <div class="product-info">
                                <h3>'.$row_prod_search['product_name'].'</h3>
                                <div class="product-desc">
                                    <h4>WOMEN'S</h4>
                                    <p>Lorem ipsum dolor sit<br />amet</p>
                                    <strong class="price">$58.99</strong>
                                    <br />
                                    <a href="index.php?add_cart='.$row_prod_search['product_id'].'"><strong >Buy Now</strong></a>

                                </div>
                            </div>
                        </li>
                    ';
            }
        }
        else{
            echo '<div class="error">Sorry There are not Products to Show it Related with your searc</div>';
        }
```

3. الدالة GET مفيدة في أشياء كثيرة فمثلاً محركات البحث يجب أن تستخدم هذه الخاصية لكي يستطيع المستخدم أن يستخدم عنوان البحث ويحتفظ به لوقت آخر ولا يقوم من جديد بكتابة الكلمة التي يبحث عنها وذلك لأن البرامترات الممررة عبرها تبقى ضمن browser history.

ثانيا POST:

كما ذكرنا تستخدم أيضاً الدالة POST لتمرير قيم الحقول المدخلة من قبل المستخدم إلى الصفحة الهدف ولكن بطريقة لا يمكن من خلالها معرفة قيمة الحقل التي تم إرسالها بواسطة هذه الدالة حيث لاتظهر في شريط العنوان للمتصفح بعد الرابط ويمكن تلخيص أهم الأماكن التي يفضل استخدامها فيها وفق مايلي:

1- غالباً مايتم استخدام هذه الدالة في عمليات تسجيل الدخول والولوج إلى قواعد البيانات.وذلك لسرية المعلومات التي يتم إدخالها كالاسم والإيميل وكلمة السر..ومن الطبيعي أن تكون هذه المعلوات سرية لذلك تستخدم في هكذا حالات الدالة POST .

2- في عمليات الدفع الإلكتروني وإدخال البيانات إلى سلة المشتريات..من الضروري أيضا أن تكون المعلومات المرسلة بين الصفحات سرية وغير ظاهرة للعلن.وهنا يجب استخدام هذه الدالة أيضا.

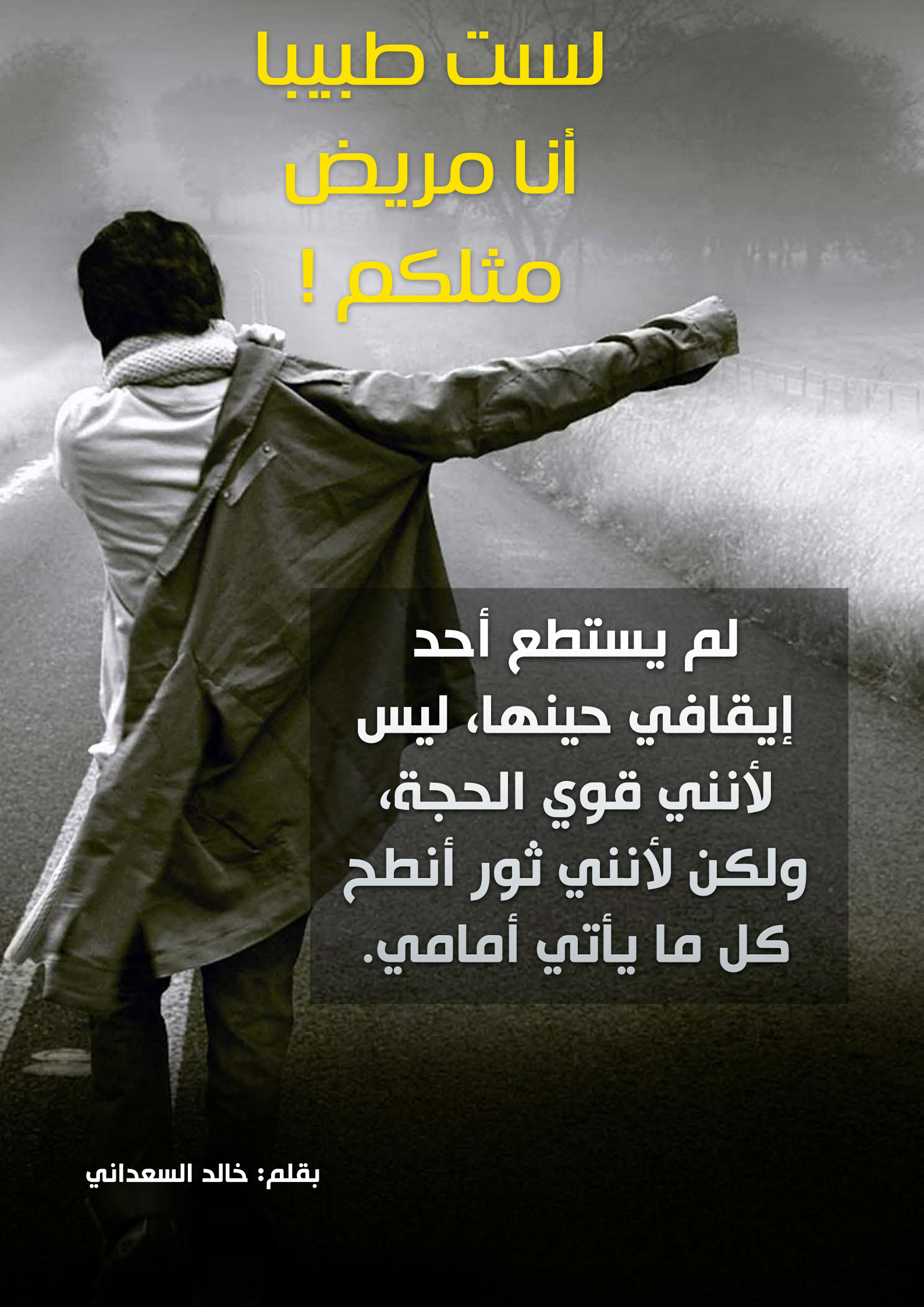
وعلى العموم لايمكن حصر أماكن استخدامات هاتين الدالتين بشكل دقيق. ولكن قد بينت أهم الأماكن التي يمكن للمبرمج استخدامها فيها والآن سنقوم بتلخيص خصائص كل واحدة بالنقاط التالية:

أولا GET:

- البارامترات الممرة عبرها تبقى ضمن browser history لأنها تكون جزء من الURL .
- لايجب استخدام الدالة GET عند التعامل مع البيانات السرية والحساسة.
- حجم البيانات التي يتم نقلها عبرها صغير مقارنة بالدالة POST حيث أن أقصى عدد محارف يمكن نقله عبرها على سبيل المثال في متصفح Internet Explorer هو:2,048 محرف. والسبب في ذلك هو أن البيانات يتم نقلها في URL وهو مقيد بطول معين.
- يمكن القول ان الدالة GET أسرع من الدالة POST

ثانيا POST:

- البارامترات الممررة عبرها لا يتم حفظها ضمن browser history.
- تستخدم عند التعامل مع بيانات سرية وحساسة ككلمات السر والمعلومات البنكية ..
- يمكن من خلالها نقل كميات كبيرة من البيانات حيث يمكن عبرها نقل ما يقارب 8 Mb من البيانات وذلك يعود لسبب أن البيانات لايتم نقلها عبر الURL كما في الدالة GET بل يتم نقلها ضمن الـ header للصفحة.



لست طبيبا أنا مريض مثلكم !

لم يستطع أحد إيقافي حينها، ليس لأنني قوي الحجة، ولكن لأنني ثور أنطح كل ما يأتي أمامي.

بقلم: خالد السعداني

لست طبيبا أنا مريض مثلكم !

خالد السعداني

أنا مريض، وأنتم الأطباء

كنت أنظر إلى الحائط القصير، فأخشى إن أنا حاولت القفز عليه أن أتعثر في الهواء فأسقط على وجهي، فتسقط مع سقوطي أسناني البيضاء الجميلة فأمضي بقية عمري أدردا بلا أسنان يتغامز علي الناس ويشير إلي الصبية بأصابعهم هازئين، تنتابني حالة من الذعر والرهاب من أشياء مستحيلة الوقوع في عرف الإنسان العادي، لكنها في عرفي أقرب إلي من أرنبة أنفي.

خوفي لم يكن مقصورا على خشية الألم من سقوطي على وجهي وأنا أقفز على الحائط القصير، بل تجاوزه ليشمل أشياء غير ملموسة، فكنت أرتاب من المكوث وحيدا مخافة الجن، وكنت أهاب الخروج ليلا مخافة اللصوص وقطاع الطرق.

كنت أكتم أمراضي ولا أبوح بها إلى أحد، حتى أمي التي يفترض أنها أقرب الناس إلي لا أبث إليها شكواي، وإنما ألوذ بالصمت والهدوء المصطنع وفي قرارتي تتصارعني الأمواج وتتقاذفني العواصف.

كنت أنظر إلى الناس كافة أنهم أسوياء، يجيدون فن الحياة، يتكلمون فيحسنون، يعملون فيتقنون، يناقشون فيقنعون، فكنت أردد دوما حينما أختلي بنفسي:

إنهم أطباء وبيدهم علاجي، علي أن أجتمع معهم، وأستفيد منهم ومن خبراتهم.

ثم أنظر إلى نفسي نظرة دونية محتقرة، وأبدأ في الجلد الذاتي دونما رحمة أو شفقة:

سُحْقًا لي من فاشل، تَبًّا لي من رعديد، ما هذا الكم الهائل من الفوبيا الذي أعيشه؟ فوبيا من الارتفاعات، فوبيا من نباح الكلاب، فوبيا من السباحة، فوبيا من الحديث أمام الناس، فوبيا من الوحدة...

أقف أمام المرآة في أحايين كثيرة، وَأَتَخَيَّلُنِي أمام الناس فأضغط على أعصابي وأصيح بشدة:
أنا مريض وأنتم الأطباء !
بحثت في الانترنت عن تفسير لحالتي، فوجدت أنهم يسمونها في علم النفس بالوسواس القهري، فتنفست الصعداء وزفرت بحرارة لأنني لست الوحيد الذي اعتراه هذا المرض، لكن سرعان ما عاد المارد الذي يسكنني ليكدر علي صفو مزاجي قائلا: الصوت الذي ينبهك ليس وسواسا إنه صوت الضمير، ويحك أترضى أن تخالف فطرتك وتتبع أهواء قوم ضلوا؟
ثم ما لبث أن أردف المارد معاتبا:
هؤلاء قوم تفرقت بهم السبل، فبدؤوا يُسَمُّونَ صوت الضمير بالوسواس القهري، أنت لست مريضا، بل هم المرضى إذْ ما بقي لصوت الضمير عليهم سلطان.
أَقْنَعَنِي المارد فأغلقت صفحات الويب التي كنت أطالعها وكلي يقين أن علماء النفس هم المرضى، وأنهم في مكانة دون مكانتي، فأخذتني غمرة الغرور للحظات، فابتسمت بكبرياء وأنا أردد في نفسي: لست مريضا، أنتم المرضى.

أنا طبيب، وأنتم المرضى

كنت أعلم أن قلقي وخوفي لم يمت، لكنني بنيت عليهما وهما جديدا، فتخيلتني طبيبا بارعا يفهم حالة البشر المرضية، وبدأت أعامل الجميع وفقا لهذا التصور الطارىء، أذكر أنني تحدثت مع والدي بشدة وبلهجة خشنة حينما وجدته يشرب الماء باردا بعد وجبة دسمة، أحسست بتأنيب الضمير ساعتها، لكنني ما لبثت أن عدت إلى غطرستي وهمست بهدوء: إنما أقسو عليه لمصلحته !
بدأ الناس يتعاملون معي بطريقة مختلفة، فصار أصدقائي يتحاشون اللقاء بي، والدي بدأ يقلق علي أكثر من ذي قبل، أما أمي المسكينة فإنني كنت أسمع همهماتها عند الدعاء فلا أميز منها إلا قولها: "اللهم رد إلينا ابننا ردا جميلا"، يعجبني هذا الدعاء فأردد خلفها: اللهم رد أمي وأبي وسائر الناس إلى صوت الحق ردا جميلا.
كنت على يقين أنني أمسك الحق في يدي، كما تمسك العجوز عكازتها، وكما يمسك الأعمى عصاه، لم أشك للحظة، أنني زدت على مرضي القديم مرضا جديدا، وكيف أعي ذلك وأنا سعيد جدا بما بلغته من نبوغ وتميز.
بدأت أرد على شيوخ الدين في صفحتي على الفيسبوك، أرد أحاديث شتى لأنها لا توافق مزاجي، أوزع اللايكات على مقاطع السخرية من الغير، لم يستطع أحد إيقافي حينها، ليس لأنني قوي الحجة، ولكن لأنني ثور أنطح كل ما يأتي أمامي.
وصلت إلى حالة هستيرية من الإعجاب بالذات صرت أرى معها كل قول غير قولي سخفا، وفي كل نصيحة تسدى إلي ضعفا.
أيها الناصحون، احتفظوا بأغراضكم لأنفسكم، فإن عقلي الكبير لا يستحمل سخافاتكم !

أيها الناصحون، احتفظوا بأغراضكم لأنفسكم، فإن عقلي الكبير لا يستحمل سخافاتكم !

مع مرور الزمن، بدأ اليأس يتسرب إلي، ثم تلاه الأرق، ثم تلاه فقدان كل لامع لبريقه، فما عاد يغويني شيء..

وصلت إلى حالة من التعب النفسي والفراغ الروحي لم أصلها من قبل..

حاول المارد مجددا أن يصور إلي هذا المرض على أنه نعمة فقال لي:

ما لك لا تستحمل سطوة العقل؟ إنما هذا التعب مرجعه إلى عقلك الضخم، وما استراح من كان له عقل كعقلك.

كاد يقنعني هذه المرة أيضا، لولا أن صداع الرأس جَرَّاءَ الأرق حال دون ذلك، فَغِبْتُ في دوامة من الأفكار كدت أن أفقد عقلي معها، فأوقفت هذه المعمعة وأنا أصيح:

لست طبيبا أنا مريض مثلكم !

ألست طبيبا، أنا مريض مثلكم

"كلنا مرضى" أهون علي من أن أكون مريضا دون العالمين، أو طبيبا على عيادة من المرضى تمتد من القطب الشمالي إلى القطب الجنوبي.

ما لي وللناس؟ سأحيى حياتي وفق الخط المحمدي، وَأَعْظِمُ به من خط !

أما خوفي من المجهول فقتلته بالإيمان بالقضاء والقدر..

وغروري وتكبري قهرتهما بالتواضع وخشية الرحمن وخفض جناح الذل من الرحمة لوالدي، وألا أمشي في الأرض مرحا.

هكذا علمني الإسلام، وهكذا انتشلتني أنامل الرحمة من براثين العذاب، فالحمد لله رب العالمين !

تجارب برمجية

كيف برمجت أول لعبة لنظام أندرويد

بقلم: خالد السعداني

تجارب برمجية

كيف برمجت أول لعبة لنظام أندرويد

خالد السعداني

لم تكن عندي خبرة سابقة مع برمجة تطبيقات الأندرويد. بحكم حقل تخصصي الذي حَتَّمَ عَلَيَّ المكوث في دائرة برمجة تطبيقات سطح المكتب وتطبيقات الويب Desktop and Web Programming. ولطالما رددت في خَوَالِجِ نفسي أن أكبر أكذوبة تحول بيننا وبين تعلم المزيد هي أكذوبة التخصص. التي تقول أن الإنسان لا يستطيع أن يجمع بين تخصصين أو أكثر. لأن الذهن سيتشتت والتركيز سيطير شعاعا. وكأن العقل البشري محدود القدرة. غير قادر على حيازة أكثر من صنف معرفي.

ألقيت نظرة على كتب التاريخ. فوجدت أن أبا جعفر الخوارزمي رحمه الله جمع بين علم الحساب. الفلك. الجغرافيا. الهندسة. والتاريخ. ثم جاء بعده أبو بكر الرازي. فبرع بدوره في شتى العلوم. فجمع بين الطب العضوي والنفسي والكيمياء والجيولوجيا والفلك والسياسة.

فأيقنت حينها أن هذه العقول الجبارة ويوجد مثلها الكثير في تاريخنا الوضاء. لم تؤمن بمسألة التخصص وإنما هزت شجرة العلم هزا عنيفا فتساقطت عليها الثمار من كل جانب. وكذلك يجب أن نكون.

كنت أنتوي أن أدخل إلى مجال تطبيقات الأندرويد منذ زمن بعيد. لكنني كنت أرجىء الأمر إلى حين. فلما احتجتها عزمت على تعلمها. ولما بدأت مسيرة التعلم أحببت أن أشاركها معكم لينتفع بها كل من له بها حاجة.

قبل أن أحدثكم عن الخطوات التي قمت بها. يلزمني أن أحدثكم عن مجال برمجة الألعاب كأحد المجالات الحيوية في البرمجة.

لما كان قطاع برمجة الألعاب قطاعا مربحا جدا ويدر على أصحابه الكثير من الأموال. تهافت عليه الناس بشدة فانقسموا إلى قسمين رئيسيين:

1. قسم يعتمد على الألعاب الجاهزة فيكتفي بالتعديل على تصاميمها وتغيير عناصرها وهو ما يعرف بعملية إعادة التشكيل Reskin.
2. قسم يعتمد على القدرات البرمجية فيقوم بصناعة الألعاب من الصفر بَدْءًا بتصميم العناصر Sprites وانتهاء ببرمجة الفئات Classes.

القسم الأول ليس مطلوبا منه أن يكون مبرمجا، إذ في الغالب يقوم بتغيير صور عناصر اللعبة أو التعديل عليها ببرامج الرسم مثل فوتوشوب، وأقصى ما يقوم به التعديل على أوامر برمجية بِنَاءً على إرشادات دون أن يلزمه من ذلك فهمها، وهذا القسم في الحقيقة بعيد كل البعد عن الابتكار أو صناعة ألعاب جديدة، لأنه في الأصل يقوم بالمحاكاة والمشاكلة فقط، مثلا يأخذ لعبة بطلها قط نطاط Jumping Cat، فيبحث عن صورة القط ويضع مكانها صورة فأر مثلا ويعطيها اسما جديدا Jumping Mouse مثلا.

أما القسم الثاني وهو القسم الواعي بما يفعله، فهو يبدأ بفكرة اللعبة أولا، ثم يقوم بتفكيكها ذهنيا ليحصل على كافة عناصرها، ويقوم بوضع تصور شامل للعبة، وبناء على ذلك يخوض عملية البرمجة بحيث يمثل كل عنصر من عناصر اللعبة فئة Class والعمليات على شكل وظائف Methods.

عموما، نحن لن نخوض في أي القسمين أفضل فالفروق بارزة لمن رزقه الله حسن البصيرة، لكن سنفترض مسبقا أنك من قُصَّادِ القسم الثاني، لأن القسم الأول أساسا ليس ملزما بقراءة كتب البرمجة أو التنقيب في الأكواد المصدرية.

إن كنت من القسم الأول، فدعني أختصر عليك الطريق وأقول لك: إن البرمجة الكائنية التوجه Object Oriented Programming مهمة جدا في برمجة الألعاب، لذلك لا تحرم نفسك من تعلم هذا النمط البرمجي النافع فهو سيفيدك في عدة مجالات برمجية.

سر قوة البرمجة الكائنية التوجه يتجلى في واقعية تعاملها مع المفاهيم البرمجية، إذ ترتكز في الأساس على تمثيل العناصر العامة على شكل فئات Classes وبالتالي يتم إنشاء نسخ من هذه الفئات ليتم التعامل معها واستدعاء وظائفها، وتسمى هذه النسخ بالكائنات Objects.

يمكنكم الاستفاضة في البرمجة الكائنية التوجه ومفاهيمها عبر مراجعة كتابنا "الشامل في البرمجة الكائنية التوجه OOP" أو مشاهدة الفيديوهات المسجلة المرفوعة على قناتنا في اليوتيوب، أو متابعة مقالات مجلة الرابط المسماة "قصة ظهور قاصمة الظهور OOP".

الآن تعالوا بنا نتحدث عن تجربتي الشخصية مع برمجة ألعاب أندرويد، بدأت أولا باختيار مصادر

التعلم. فوقفت على كتابين باللغة الانجليزية بهما من الخير الشيء الكثير. أما الكتاب الأول فاسمه Beginning Android 4 Application Development وهو يضم تقريبا كل ما ستحتاجه لإنشاء تطبيقات الأندرويد. والكتاب الثاني اسمه Android Game Programming By Example وهو يعلمك كيف تنشىء ثلاث ألعاب من الصفر بشرح متدرج سلس.

الكتاب الأول لم أغص فيه بتمعن وإنما ألقيت عليه نظرة تفحصية فسرني جدا محتواه. فأرجأت قراءته إلى الحاجة. لأن عناصره المعرفية أشبه ما تكون بحلول عملية لوضعيات معينة ولست في حاجة إليها حاليا. مثلا:

1. التعرف على ماهية Activity و Fragment و Intent
2. التحكم في طرق عرض التطبيق على أجهزة أندرويد
3. استخدام الأدوات الأساسية مثل textView وغيرها..
4. استخدام الأدوات المتقدمة مثل ListView وغيرها...
5. التعامل مع البيانات عبر الملفات وقواعد البيانات
6. كيفية إجراء المراسلات SMS Messages and E-mails
7. التعامل مع تحديد المواقع الجغرافية والخرائط
8. التعامل مع الشبكات Networking عبر تحميل البيانات الثنائية والنصية
9. استهلاك خدمات الويب Web services
10. نشر التطبيق ورفعه على متاجر التسويق
11. التعرف على بيئة Eclipse المتاحة لتطوير تطبيقات الأندرويد
12. التعرف على الجهاز المحاكي Emulator لاستعراض تطبيقات الأندرويد

الكتاب كما عرضته لكم باختصار كتاب ذو خير عظيم لمن شاء أن يتمرس في تطبيقات نظام الأندرويد.

أما الكتاب الثاني الذي اعتمدته دليلا لبرمجة الألعاب فهو كتاب Android Game Programming By Example.

الكتاب يطرح مواده على شكل دروس تطبيقية لو أداها صاحبه على شكل فيديوهات لكان أشد نفعا وأكثر إفادة. لكن مع ذلك لعب أسلوبه الواضح في الشرح المتدرج دورا مهما في جعل تطبيق محتويات الكتاب عملا سهلا ومفهوما.

لم أجد صعوبة بالغة في تطبيق مقتضيات الكتاب بحكم أن الخطوات البرمجية المنجزة ليست اختراعات جديدة وإنما مجموعة من الأوامر التي ترتكز بالأساس على البرمجة الكائنية التوجه.
لذلك أعود وأؤكد مجددا على الإخوة الأفاضل والأخوات الفاضلات المقبلين على برمجة ألعاب أو تطبيقات أندرويد، أن يتمرسوا أولا في البرمجة الكائنية التوجه وأن يتقنوا أساليبها، ثم بعد ذلك سيهون كل شيء.

أما لغات البرمجة التي أنصح بها، فإنني أشير بالدرجة الأولى إلى لغة جافا بحكم قوتها ومجانيتها ووفرة مصادر تعلمها وتوافقها التام مع نظام أندرويد، أو اللغات المتقدمة الأخرى كلغة سي شارب وما حذا حذوها.

في الختام، أرجو من الله العلي القدير أن تكونوا قد استفدتم من تجربتي المتواضعة، وأن تتضح السبيل أمامكم، إن كان كذلك فبادروا فورا باقتحام هذا المجال دون أن تغفلوا للحظة أن تستحضروا طاعة الله عز وجل في تطبيقاتكم وألعابكم وأن تحرصوا كل الحرص على جناحي السلامة اللذين يضمنان للمسلم توازنه: نعم عن الحرام والحلال أتحدث.

سبيل النجاح

"

ومن لم يخطط لحياته عاش في مخططات الآخرين

"

ريما علي المصعبي

النجاح: هو رمز على جبين العظماء، وهوبصمة الأنبياء والرسل.

متطلباته:

الرغبة: هي النية الصادقة والمؤكدة والقوية في تحقيق ما تريد عملا بقول النبي الله صلى الله عليه وسلم:((إنما الاعمال بالنيات وإنما لكل امرئ ما نوى)).

فان امتلكت نية صادقة في الحصول على شيء فاعلم وتيقن بأنك ستحققه بإذن الله تعالى وتوفيقه، وحتى تولّد هذه الرغبة وتحافظ على فتيلها مشتعلا:

أولا: اعرض على نفسك هذا السؤال: هل أنت مقتنع بالوضع الذي أنت فيه؟

فكر قبل الإجابة وانتبه!!!

فإنك إذا قلت: نعم أنا مقتنع بالوضع الذي أنا فيه فلن تتحرك إرادتك مطلقا وبالتالي ينبغي أن تزيل عنك هذه القناعة الخادعة، فاجتهد الآن في إقناع نفسك بعدم الرضا وأنه يجب عليك العمل للتغيير إلى الأفضل واجعل هذا الامر تحديا أمامك لإثبات الوجود وهاجسا ملازما يراودك في اليوم والليلة.

ثانيا: حدد نقاط ضعف إرادتك وأسبابها،(البيئة، الأصحاب، الفراغ، النفس اللاهية،..) ثم فكر في الأرباح التي ستجنيها بعد الخسائر التي أنت فيها الان، فكّر في العواقب، فإن التفكير في العواقب دأب العقلاء.

ثالثا: بعد أن حددت ضعف إرادتك، فكر في طرق تعينك على التخلص من هذه الاسباب (تغيير البيئة، إشغال النفس بأمور نافعة كالقراءة، أو الكتابة، أو الرياضة، أو الزيارات، أو غيرها من النشاطات التي تناسب وضعك).

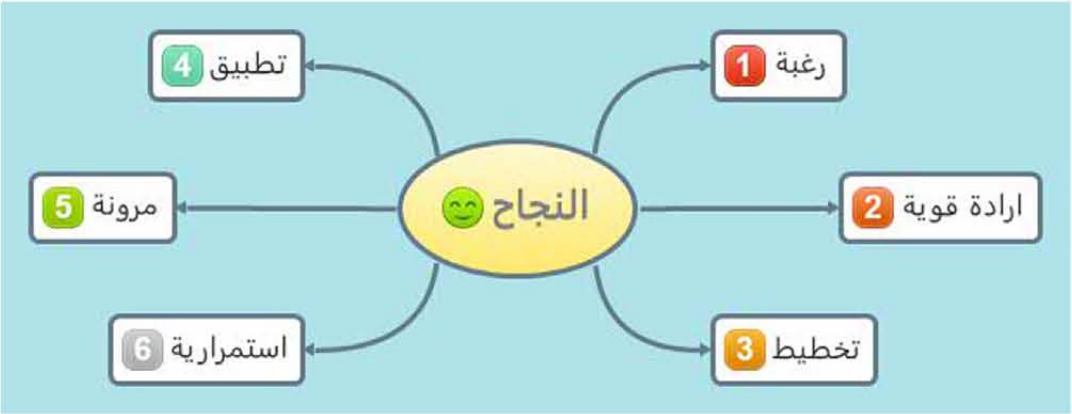
وبعد أن تأكدت من قوة رغبتك في تحقيق أهدافك، ابدأ فورا في التخطيط فمن لا يخطط يسير ولايعرف إلى أين هو ذاهب ومن لم يخطط لحياته عاش في مخططات الآخرين.

انتهيت من التخطيط انتقل إلى التطبيق وهو مرحلة كشف الجادين من الخائبين فبدون التطبيق: الأحلام والرغبات حبر على ورق وهو ترجمة الحلم الى حقيقة وواقع.

وهنا يجب عليك أن تتحلى بالمرونة ويقصد بها التعديل والتجديد المستمر بما يخدم أهدافك وبما يناسب روح العصر فليس من العدل إذا اوصدت الابواب أمامك ان لا تفكر في خيارات اخرى.

وأخيرا حذاري من التوقف والانقطاع، فقد قال رسولنا الكريم صلى الله عليه وسلم ((أحب الاعمال إلى الله أدومها وإن قل)).

ولاتَنْسَ أخي الحبيب أن النجاح هو أن تنتصر وتتغير من الداخل وليس النجاح في وفرة المال والشهرة والمناصب وإن كانت مطلوبة، ثق بنفسك وقدراتها وامْحُ كلمة فشل من فصول حياتك واجعل نجاحك الدنيوي يأخذ بيدك لنجاح الآخرة والفوز بجنة عرضها السموات والأرض.



ريما علي المصعبي

تكنولوجيا

السلسة التعليمية في التطور التكنولوجي

محمد حوتري
فلسطين

الجزء الثاني

السلسة التعليمية في التطور التكنولوجي

تطرح هذه السلسة إن شاء الله بشكل مستمر إن استطعنا بحيث سيتم من خلال هذه السلسة طرح مواضيع التطور التكنولوجي مع الشرح العملي ان لزم الأمر.

محمد حوتري

سنتطرق لمواضيع عدة منها :

- الالكترونيات الحديثة والقديمة
- المكونات الالكترونية
- الدارات الالكترونية الكهربائية
- الدارات الالكترونية الرقمية
- الدارات الالكترونية القابلة للبرمجة "المتحكمات الدقيقة"
- وغيرها

وقد يتم شرح الحواسيب الجديدة واللوحات التطويرية مثل الراسبيري باي وغيرها اذا شاء الله وأبقانا أحياء ^_^

باختصار مراحل التطور التكنولوجي الكهربائي بدأت حين تم اختراع الكهرباء بواسطة العالم الأمريكي من أصول هولندية "توماس أديسون " وكانت ثورته باختراع بطارية تقوم بتشغيل مصدر مشع للضوء "لمبة" ثم توالت الاختراعات الكهربائية في ذاك العهد من أديسون وغيره من العلماء والمهندسين .

ثم جاء عصر التكنولوجيا الذي قلب العالم رأساً على عقب: ثورة الترانزيستور

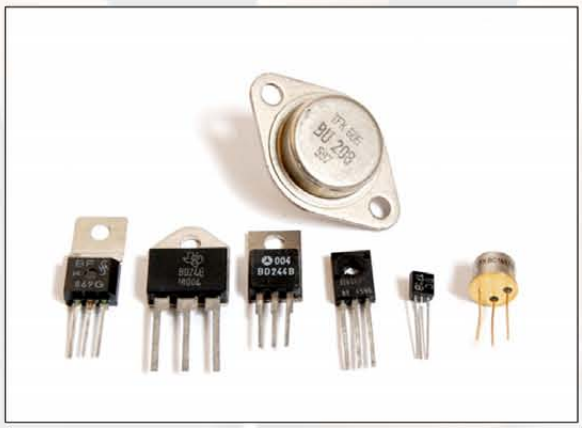
الترانزيستور الذي عمل به بشكل جدي ثلاث علماء نجحوا في اطلاقه والعمل به هم "والتر براتن" , " جون باردين " , "وليام شوكلي " وكان من قبلهم "جوليس ادجر " أول براءة اختراع للترانزيستور في كندا عام " 1925م " وكان هذا الترانزيستور شبيه بالترانزيستور تأثير المجال "FET" ولكنه لم يصدر أبحاثا او تجارب عملية واقعية ثم بعد ذلك في عام "1934م" قام الألماني " أوسكار هيل " بتسجيل براءة اختراع لترانزيستور شبيه بسابقه !

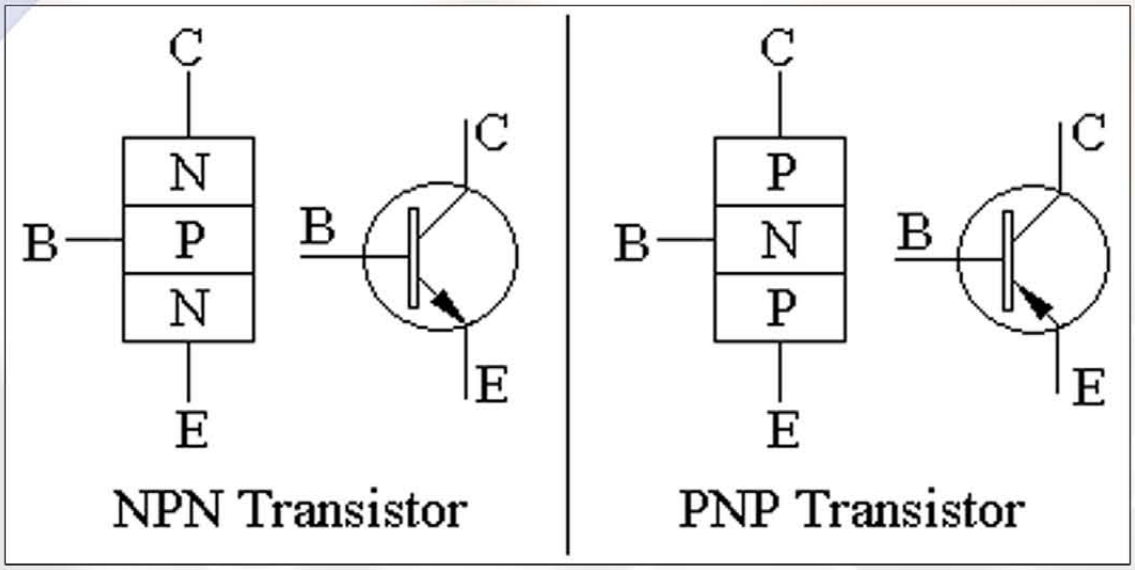
وفي عام 1942"م" قام , "هبرت مارتين " بعمل تجربة بما يسمى "الديو دايو " الوصلة الثنائية المزدوجة أثناء العمل على لاقط بنظام رادار دوبلر .

في عام 1947م قام "جون باردين" و"والتر براتين"في معامل "AT & T bell "في الولايات المتحدة الأمريكية بملاحظة انه عند توصيل مصدر كهربي على بلورة من الجرمانيوم ان الطاقة الناتجة أكبر من طاقة المصدر الكهربي الداخلة وقد قام "وليام شوكلى" بمعرفة السبب في ذلك وعلى مدار شهور قليلة عملوا على التوسع الكبير لعلوم أشباه الموصلات وقد جاء اسم الترانزستور من الكلمة الإنجليزية "Transfer resistor" التي تعني ناقل المقاومة.

في هذه الصورة قاموا بتصميم نفس اول ترانزيستور عملي:

في هذه الصورة أشكال عدة للترانزيستورات المستخدمة حاليا:





سيتم شرح الترانزيستور بالتفصيل والصور العملية في الجزء التالي ...

دمتم بخير

التفكير البرمجي

البرمجة ليست أكوادا جامدة تؤدي وظيفة معينة، بل البرمجة فكْر، و تحويل الفكرة إلى شفرات و أكواد أهون من هوان الدنيا على الله عز و جل.

صوفي عباس

أردت في بداياتي عمل برنامج تعليم الانجليزية للأطفال، يعرض في إحدى نماذجه عشر صور بشكل عشوائي غير متكرر تتغير مع كل اختبار جديد، و في "الغير متكرر" تكمن المشكله، حيث إن دالة الأرقام العشوائية تكرر الارقام العشوائية، و ليس من المنطقي ان تعرض الصوره العشوائية مرتين !! ما الحل ؟؟ وجدتها إنه سماحة العلامة حسنة الزمان غير الثقة و لا الثبت: جوجل، هرعت إليه فلم أجد الحل، إنما اغرقني في خوارزميات تنتج ارقاما من أربع خانات و أكثر و ما أريده هو عشرة أرقام تقع في المدى من واحد إلى خمسين بشكل فريد و عشوائي، فلما خاب رجائي في جوجل، لجأت لمن قطع أعمارنا في بحاره هباءً، إنه الفيس بوك، انضممت الي عشرات المجموعات المختصة في البرمجة، كتبت منشورا طلب المساعدة، فتجد الناس أحد رجلين من لم يفهم سؤالك أصلا و من لم ينتبه لكلمة "فريد" يحاول مساعدتك، فتبدأ تشرح له بغيتك، و تجده يسألك و يستفيض في السؤال، فينقلب الحال رأسا على عقب، بعد ما أردت الغوث و العون تجد نفسك تساعد من مفترض فيه أن يساعدك، فتبوء بضياع الحماس و الوقت، أما المحترفون فقليل وقتهم، و إجابتهم تقتصر على الأساسيات للمبتدئين.

بعدما يخيب مسعاك تلجأ الي الحل الامثل ألا و هو ترك المسألة لعل الله عز و جل يحدث بعد ذلك امراً. و قد كان. التقيت قدرا بمبرمج سوري متخصص في الجافا. فسألته غير منتظر الجواب. فلا الرجل معه حاسوب. و لا أنا اعرف أي شيء عن الجافا (في ذلك الوقت) .

فصك مسامعي بكلام غيّر مساري في البرمجة قالي (أنا أعطيك الفكرة و أنت صغً الكود) ففكرت (انا أصوغّ الكود ؟؟!!) و مع ذلك قلت (قل فكرتك) فقال و نعم ما قال:

- أنت تريد رقما عشوائي و فريدا يعني غير متكرر

- نعم

- عشوائي في المدى بين 1 الي 50 أمر سهل و لكن كيف نجعل الرقم فريدا أو غير متكرر؟؟

- كيف ؟

- تعمل مصفوفه تحتوي علي الارقام من 1 الي 50 . ثم تعمل داله تولد عشرة أرقام عشوائية من المدى و احد إلى خمسين و كل رقم عشوائي تستخرجه من المصفوفه تستبدله بصفر. حتى إذا اردت توليد رقم آخر لا يكون الرقم الذي ولدته في المدى..

- !!!!!

هرعت إلى الفيجوال ستوديو و عملت البرنامج و ياللهول فعلتها فعلتها ! او بالاحرى فعلها صديقي المبرمج السوري. ما استفدته من هذه التجربة ليس شفرة الرقم العشوائي الفريد. لكن الفكرة. ليس فكرة توليد رقم عشوائي. و لكن فكرة أن البرمجة ليست أكواد جامدة تؤدي وظيفة معينة. بل البرمجة فكْر. و تحويل الفكرة إلى شفرات و أكواد أهون من هوان الدنيا على الله عز و جل.

حال المبتدئين هو ذلك. يتعلم الأساسيات و تجد من لم يتعلمها أصلا و يبحث عن الاكواد الجاهزة . مقدرا سعادته عند عمل برنامجه بدون أخطاء و هو أمر ضروري لتحفيزه في البداية. و لكن ليس هذا سبيل صنع مبرمج حقيقي....

أدبيات فن تصميم المواقع

خطوات تصميم وبرمجة مواقع الويب

باسل يحيى زعرور
مدير شبكة المنهل: http://111000.net

العديد من المواقع العربية حاولت جاهدة تلبية الطلب المتزايد على دروس تعلم تصميم المواقع وتعددت المستويات , فمنها من يقدم دروسا مبسطة في قسمhtml ومنها من يقدم دروسا متوسطة المستوى لإحدى لغات البرمجة والكثير الكثير ..

ولكن قلة من الموقع تطرقت لموضوع الدخول في تصميم المواقع من البداية بالطريق الصحيح , فلا أخفيكم بحكم اختصاصي بهذا المجال أرى أن البداية الخاطئة به قد تؤدي الى نتائج لا تُحمد عقباها على المتعلم من كره لهذا المجال أو خوف منه .

وفي هذه السلسلة من المقالات سأعتمد إن شاء الله أبسط أسلوب لإدخال القارئ في هذا المجال بحيث يستطيع وبعد فترة وجيزة من قراءة المقالات أن يقرر هل يناسبه هذا المجال أو لا وبأي قسم يود أن يختص. وسأحرص دائما من جهتي على تزويدكم بالمواقع التي تفيدكم في كل مرحلة .

لنبدأ على بركة الله وتوفيقه.

من أين وكيف تبدأ في تعلم تصميم المواقع

قبل أن نجيب على هذا السؤال سنشرح مراحل تصميم أي موقع على الانترنت بشكل موجز قبل الدخول بالتفاصيل لاحقا.

مراحل تصميم مواقع الانترنت :

المرحلة الأولى : التخطيط لبنية الموقع

في هذه المرحلة يتم رسم المخطط ولو على الورق لأقسام الموقع كمعلومات وخدمات حسب طلب الزبون صاحب الموقع وهي مرحلة لن نتوقف عندها كثيرا

المرحلة الثانية : تصميم شكل الموقع وألوانه

وهذه المرحلة من أهم المراحل ففيها يتم تصميم الشكل الخارجي للموقع والصفحات الداخلية ويستخدم فيها أحد برامج التصميم وأشهرها Adobe Photoshop بإحدى إصداراته أو برامج تصميم أخرى مثل (Adobe Fireworks – Adobe Illustrator) وغيرها.

وعادة في شركات تصميم المواقع يتم إسناد هذه المهمة إلى قسم التصميم لأنها مهمة تصميمية بحتة لاتحتاج أي معرفة بلغة html أو php أو اي شئ عن تصميم المواقع

المرحلة الثالثة : تحويل التصميم الى صفحة html

حيث يتم نقل التصميم من صورة جامدة إلى صفحة html من دون بيانات وهذا يتطلب من مصمم الموقع ان يتعلم لغة html واستعمال الأنماط CSS وعدة أمور أخرى، حتى تكون النتيجة كصفحة html مطابقة للتصميم الاساسي الذي بدأنا به وطبعا نحتاج في هذه المرحلة استعمال Adobe Photoshop للتعامل مع التصميم الاساسي وتصدير عناصره الى صفحة html النهائية.

والأهم أننا سنتعامل مع برامج إدارة المواقع وأشهرها Adobe Dreamweaver وكذلك Frontpage قديما وبرامج أخرى

المرحلة الرابعة : وضع البيانات في الصفحات وربطها

وخلال هذه المرحلة وباستخدام برامج إدارة المواقع يتم وضع البيانات بالصفحة بإحدى الطريقتين :

- يدويا : وذلك عن طريق مصمم الموقع حيث يتم نسخ البيانات من برنامج Word مثلا ولصقها في الصفحة فورا وهكذا ينتج لدينا مايسمى موقع ثابتة static وهو ماسنشرحه لاحقا
- ديناميكيا : وذلك عن طريق لغات البرمجة مثلPhp و Asp وقواعد البيانات مثل mysql, حيث يتم تهيئة الصفحة لاستقبال البيانات عن طريق لغات البرمجة ويتم تخزين البيانات في قواعد البيانات , وبهذا نحصل على موقع ديناميكي Dynamic Site

المرحلة الخامسة : تجميل الموقع :

وفي هذه المرحلة يمكن إضافة ميزات على الموقع مثل شريط إخباري أو صور متغيرة أو خدمات تفاعلية أكثر مثل المراسلة والتعليقات , وتتطلب هذه المرحلة ايضا إتقان لغات البرمجة وقواعد البيانات والفلاش وبعض التقنيات وسنشرحها لاحقا ان شاء الله

المرحلة السادسة : رفع الموقع ونشره على الانترنت

وهذه المرحلة تتضمن حجز اسم للموقع (الدومين Domain) وربطه بمساحة على الانترنت ونشر الموقع وإعداده للعمل بشكل صحيح وتتطلب من المصمم معرفة مفاهيم اساسية سنشير لها في حينها ان شاء الله.

وهكذا عزيزي القارئ تكون قد تعرفت بشكل سريع على مراحل تصميم المواقع وبرمجتها ولاحقا سنشرح كل مرحلة منها بالتفصيل مع مصادر باللغتين لتعلم كل مايحتاج لاتقان هذه المرحلة تحياتي لكم

أَيْقِظْهَا فَإِنْ عَادَتْ لِلنَّوْمِ فَنَمْ !

لست ألومك على ضُعْفِ هِمَّتِكَ، لكنني ألومك على تَجَنِّيكَ على حَقْلٍ دَخَلْتَهُ فَتَوَهَّمَكَ الناس من أَعْمِدَتِهِ فَلَمَّا نَكَصْتَ على عَقِبَيْكَ استهانوا بما دعوتهم إليه، فَخَلَطُواْ بين دعواك وهواك، فرفضوا كل ما جِئْتَهُمْ به، ونهجوا معك نهجَ الناكصين الْمُحْجِمِين.

بقلم: خالد السعداني

أَيْقِظْهَا فَإِنْ عَادَتْ لِلنَّوْمِ فَنَمْ !

الإصلاح بين الإرداة والضوابط

خالد السعداني

سنحيى على هذه الأرض حتى نرى من سَيَتْعَبُ أَوَّلًا. هُمْ يَظُنُّونَ أن الضرب بالمطرقة على هاماتنا سَيُثْنِينَا عن تبليغ الرسالة. لا والله لن يكون لهم ما يريدون. قد وضعنا أول خطوة في درب الإصلاح ولن نقتلعها منه إلا حينما يقتلع عزرائيل أرواحنا من بين جنوبنا.

أَتَمَزَّقُ لَوْعَةً وَأَتَقَطَّعُ حُرْقَةً على حال هذه الأمة الكليمة. لا يهنأ لي بال ولا يغمض لي جفن. أكابد الهموم في وضح النهار. وأصارع الأرق في بطن الظلام.

شدتني الحيرة في بداية عهدي بهذا الشعور. كيف سأتصرف؟ بل ولماذا علي أنا تحديدا أن أتصرف؟

فكرت مليا في الأمر. فوجدت أن للإصلاح وجهان:

أولهما إصلاح النفس ثم الأقرب فالأقرب..
وثانيهما إصلاح المجتمع في ضربة واحدة...

الوجه الأول لم يناسب حماسي واندفاعي لأن الطيش بلغ مني كل مبلغ. والوقت ليس في صالحي. على الأقل هكذا يُخَيَّلُ إلي فَنَحْنُ المصلحون نموت صغارا. نفسي هَيِّنَةٌ علي وإصلاح عيوب الغير أولى من إصلاح عيوب نفسي. ثم إنني ألعب دور المصلح فلماذا سأحتاج الصلاح أصلا؟ متى كان لزاما على الْبَنَّاءِ أن يملك قصرا؟ ومتى كان واجبا على الخياط أن يحوز مصنعا؟

قياس قد يبدو فاسدا بعض الشيء. استشعرت ذلك فعلا لكن غمرة الحماس وطيش الإصلاح الاندفاعي منعاني من الإفحاء والتدقيق.

بدأت عملية الإصلاح وليس لي في الصلاح الذي أدعو إليه نصيب، وَخْزُ الضمير غاب جراء قمعي المستمر له، فكان يقول لي: ويحك، أتدعو الناس إلى شيء لست منه في شيء؟ ألا ترى أنك تلعب دور الحرباء التي تتلون بكل لون لِتُوهِمَ الرائي وهي ليست من ذلك اللون في شيء؟

حاولت إسكات الضمير بالقوة لكن صوته كان منبعثا من بين دخان استعلائي:

(ما أقبح منظر الحرباء حينما تنصح غيرها بالثبات على لون واحد، و شَرُّ الناس من حذوا حذوها.
أولئك الذي رآهم نبي الرحمة صلوات ربي وسلامه عليه ليلة الإسراء و شفاههم تقرض بمقاريض من نار.
الذين قال فيهم رب العزة جل جلاله: "أتأمرون الناس بالبر وتنسون أنفسكم وأنتم تتلون الكتاب أفلا تعقلون؟ ".
حَاذِرْ صحبة هؤلاء القوم فإن شرهم مُعْدٍ، وإن كان فيك شيء من دخلهم فتوقف عن سرد المواعظ وإسداء النصائح فإنك أولى بها من غيرك.
وكم من قول يسرك تعميمه وهو حجة عليك لا لك، إذ لا فضل في علم ينقصه العمل به فكيف إذا ناقضه وعاكسه؟
وتذكر:
تتحسن الأحوال..
حينما تكون الأعمال مرآة للأقوال)

كلام جميل فعلا، لكنك أيها الضمير لا تقدر حجم الرسالة، ولا ثقل المسؤولية، لو انْشَغَلْتُ بإصلاح نفسي فمن سَيُصْلِحُ المجتمع والإفساد شغال على قدم وساق؟

ومضت أعوام كثيرة وأنا أَسْلُكُ كل السبل للإصلاح، أَلَّفْتُ كُتُبًا، ألقيت محاضرات، مشيت في الأسواق وفي الشوارع وَاعِظًا، لكنني لما جلست مع نفسي أتفحص حصيلة جهدي، وجدت أنه خَالٍ من البركة، فكل الأعمال التي قدمتها لم تُثْمِرْ، الناس هم الناس، والعيوب هي العيوب.

فَنَكَصْتُ على عَقِبَيَّ، وأحجمت عن هذه الأعمال النبيلة، وَغَضَبْتُ على هذه الأمة التي ترفض الإصلاح، لقد حاولت إيقاظها من نومها فلم تستجب، لقد أَدَّيْتُ ما عَلَيَّ، اللهم هل بلغت اللهم فاشهد !

مرارة الحسرة لم تفارقني للحظة، و لم يكن يُعَزِّينِي عن ذلك إلا صوت المارد الذي يسكنني:

أيقظها، فإن عادت للنوم فَنَمْ !
قد أَدَّيْتَ الشطر الأول، فلم تستيقظ، فما عليك إلا الانخراط في زُمْرَةِ النائمين !

كان كلام المارد حُلْوًا، وقع على قلبي كقطرة الماء البارد على حلق اللهثان، لكن شيئا ما ليس على ما يرام، صوت الضمير يعلو شيئا فشيئا، بدأت أتحاشاه لكنه أبى إلا أن يزداد علوا حتى مسك مجامع قلبي، فكان من جملة ما قاله الضمير:

ها قد انضممت إلى حلف المتخاذلين بعد أن نبهتك منذ مدة إلى أن وجه الإصلاح الأول هو الأجدى، كان يليق بك أن تبدأ بإصلاح نفسك أولا ثم تُثَنِّي بإصلاح الأقرب فالأقرب، لكنك أبيت إلا أن تقتحم هذا المجال الْوَعْرَ، فتسيء إليه وإلى أهله.

كيف سينظر الناس الآن إلى دعوتك بعد أن تخليت عنها وركنت إلى حلف المتخاذلين؟ كيف ستستيقظ هذه الأمة وقد حجزت لنفسك منذ أول عهدك بالإصلاح سريرا إلى جانب نُوَّامِهَا؟

قد فضحت خبيئتك، كنت أعلم من الأول أنك تحاول أن تزيح ثقل المسؤولية عن عاتقك فسلكت درب الطائشين، وهذه قطوف عدم الإخلاص !

لست ألومك على ضُعْفِ هِمَّتِكَ، لكنني ألومك على تَجَنِّيكَ على حَقْلٍ دَخَلْتَهُ فَتَوَهَّمَكَ الناس من أَعْمِدَتِهِ فَلَمَّا نَكَصْتَ على عَقِبَيْكَ استهانوا بما دعوتهم إليه، فَخَلَطُواْ بين دعواك وهواك، فرفضوا كل ما جِئْتَهُمْ به، ونهجوا معك نهج الناكصين الْمُجْرِمِين.

كان الصوت هذه المرة أكبر من صوت المارد، بل وأكبر من صوت الطيش الذي رزأني أعواما من عمري، آن لي - وإن كان الأمر متأخرا - أن أبخع لصوت الحق وأخضع له، فالإصلاح كقطرة الزيت على الثوب، تبدأ من حيث وقعت لتمتد إلى ما حولها، أما الإصلاح الاندفاعي فشبيه بالريح العاتية، تذري معها كل شيء حتى تَسْكُنَ فَتُفْنِي نفسها.

قهوة الرابط

فضاء للاستراحة

عجائب الدنيا الثلاث

قال أحد الحكماء:
عجبت في الدنيا من ثلاث،
أما الأولى فبرنامج أنجزه فيعمل عندي بنجاح..
وعند الاختبار أمام العميل يصيبه الكساح...
وأما الثانية فمبرمج يكتب الكود من غير تحليل...
ثم يتساءل لماذا نجاح مشاريعي مستحيل...
وأما الثالثة فلغة برمجية زاولتها دَهْرًا...
فَرُفِعَ الدعم عنها وهلكت دونها قَهْرًا...

مصباح علاء الدين

في إحدى شوارع أنقرة كان هنالك مبرمج يتسكع بين الأزقة فوجد مصباحا يشع، فتلقفه من الأرض وبدأ يمسح ما علق به من غبار، وفجأة خرج منه مارد أزرق وهو يقول:
– شبيك لبيك، طلباتك بين يديك.
اندهش المبرمج وقال في خوف: من أنت؟
هدأ المارد من روعه قائلا: أنا خادمك، اطلب ما تشاء وعلي التنفيذ.
لما سمع المبرمج كلمة "التنفيذ" ضحك ساخرا وهو يقول: وماذا لو حصل معك bug في زمن التنفيذ ؟
لم يتقبل المارد هذه المزحة وغضب غضبا شديدا، ثم دخل إلى المصباح وهو يقول: أعطينا الأحمق كراعا، فطلب ذراعا.
فندم المبرمج أشد الندم على تضييعه هذه الفرصة الذهبية.

مبرمج في العصر العباسي

خرج الأمير ذو البشائر يتصيد في إحدى الغابات مع بعض حاشيته، فلما كان ببعض المروج أضاع سبيل العودة، فالتمس مخرجا وبعد طول مجاهدة لقي مبرمجا يتفيأ ظل شجرة مع حاسوبه.

فصاح به الأمير بشدة:

- يا غلام أرشدنا إلى مخرج فإنني منذ ساعتين عالق هنا.

وضع المبرمج السماعات إلى جانبه ورد باقتضاب:

- ساعتين فقط وتكاد تنفجر غضبا، فما قولك في من لازم مكانه هذا لثلاثة أيام محاولا أن يصلح خطأ ولم يقدر عليه؟ حتى مبرمجي StackOverFlow عجزوا عن المساعدة.

لم يفهم الأمير شيئا فابتسم ببلاهة وهو يقول:

- لم أفهم قصدك لكن على كل حال كان الله في عونك.

فرد المبرمج ببرود:

- مع السلامه يا شيخ !

ففر الأمير فاه وهو يقول:

- ثكلتك أمك، أتدري من أنا؟

فأجاب المبرمج:

- كن من تشاء، فأنت مجرد Object من كلاس الإنسان، تم إنشاؤك ك Instance يوم ولادتك وقريبا سيقدر لك Finalize ، فتواضع يرحمك الله.

ثم واصل بنفس الهدوء:

- تعال إلى جانبي أفتح Google Map ثم أرشدك إلى خربتك.

ضحك الأمير بقوة قائلا:

- لله درك من فصيح، لم أفهم من كلامك شيئا لكنه من غير شك قول حكيم.

من مسخرة إلى مفخرة

مقدمة:

بنظرات زائغة يملؤها الشك ينظر إلى الناس من حوله، لماذا هم هكذا معه؟ لماذا يصفونه بالغبي؟ هل لأنهم لا يفهمونه فظنوه أبلها؟ أم لأنه فعلا كذلك؟ الجواب على هذا السؤال وحده يكفي لتحديد هل سيكون هذا الشخص ناجحا أم فاشلا. إن كان جوابه بأنه ليس غبيا، وإنما قدرته الذهنية تفوق تصور هؤلاء الهازئين فقد أحسن صنعا وخطا أول خطوة في درب النجاح. وأما إن كان جوابه بأنه فعلا غبي والناس صادقون حتما فيما يصفونه به فقد أساء صنعا ووضع أول قدم في هوة السقوط المدوي.

مراحل بناء الثقة المفضية إلى النجاح:

صحيح ما يقولونه عني..
قد يكون صحيحا ما يقولونه عني..
بل ليس صحيحا أبدا ما يقولونه عني...
أنا مفخرة !

مراحل الجلد النفسي المفضي إلى الفشل:

ليس صحيحا أبدا ما يقولونه عني...
قد يكون صحيحا ما يقولونه عني...
بل هو صحيح ما يقولونه عني...
أنا مسخرة !

الخاتمة

الإنسان بين الموازين الثابتة والأعراف

الحمد لله ابتداء وانتهاء وبعد:

فقد من الله علينا بإكمال هذا العدد الثاني من مجلة الرابط، والذي حباه المولى تبارك وتعالى طابعا خاصا يتجلى في تنوع موضوعاته، وفي مبادرة الإخوة والأخوات الأفاضل إلى المشاركة في وضع أولى لبنات هذا الصرح المعرفي.

من المعهود أن تكون الكلمة الخاتمية ملخصا لما تقدم اختياضه من مقالات وموضوعات، لكننا لن نلتزم بذلك لأننا نثق بقرائنا الكرام، الشغوفين بقراءة كل فقرة من فقرات المجلة، فنحن نحرص على أن تكون كل جملة مخطوطة على هذه الصحائف الالكترونية مسمارا نباشا يترك آثارا إيجابية في فكر المتلقي.

لذلك لن نلخص ما سبق، وإنما سنتحدث عن موضوع آخر يتمثل في: الإنسان بين الموازين الثابتة والأعراف.

الموازين الثابتة هي تلك البنود التي جاءت بها شريعة الرحمن لنقل الإنسان من حضيض الجهل والتدمير إلى قمة العلم والتعمير، والأعراف هي تلك البنود المتغيرة التي يتعارف عليها بنو البشر عامة، أو قوم دون قوم حسب حصرية الثقافة وحدودية التقاليد والطقوس.

أقول وبالله التوفيق، أن الإسلام لم يأت ليضع للإنسان قالبا ليقيد حركاته ويشل أفكاره، وإنما جاء لينقذ الإنسان من براثين الجهل وأوحال الشهوة المطلقة ويضعه على طريق الفلاح (المحجة البيضاء) عبر ترشيد الشهوات، وتنظيم الأفكار والأقوال والأفعال وتقسيمها إلى نافع وغير نافع (حرام وحلال).

فأي شيء من الأعراف جاء مخالفا لرؤية الإسلام فلا ينبغي التمسك به تحت ذريعة أنه من تقاليد البلد أو لأن الأمم اتفقت عليه.

لن أخوض في التفاصيل فهي كثيرة للأسف، لكنني سأكتفي بتلخيص رسالتي فأقول: أن الله عز وجل سيحاسبنا وفق الموازين الثابتة التي جاء بها الإسلام وليس وفق أعرافنا وعاداتنا وتقاليدنا المتغيرة والتي تجانب في كثير منها مقاصد الإسلام.

لذلك وجب علينا أيها الإخوة الكرام/ الأخوات الكريمات، أن نتبين جيدا هذه الأعراف فإن كانت موافقة للإسلام فمرحبا بها وإلا فعرض الحائط.

دام لكم البشر والفرح.