

مقدمة عن تطوير الويب باستخدام Django

كيف يعمل Internet

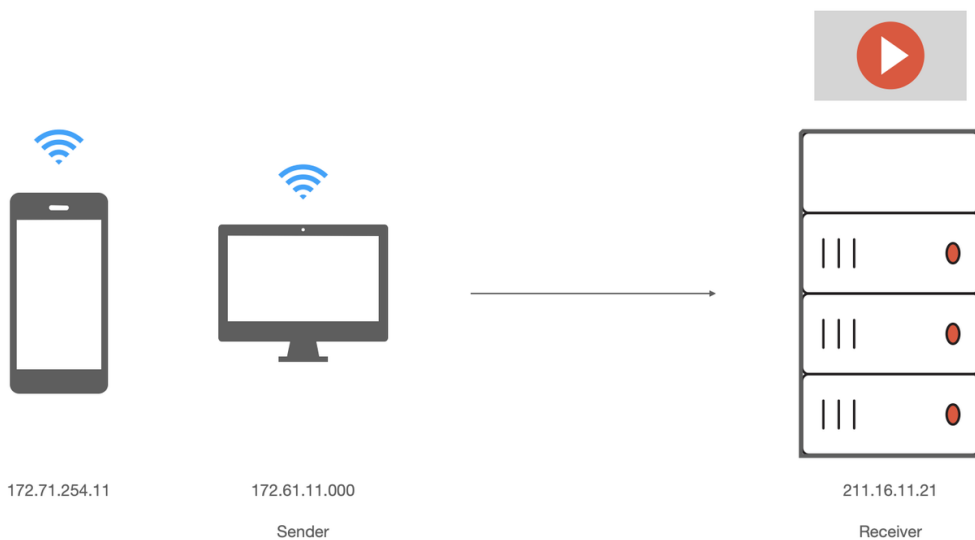
لنفترض أنك تريد الوصول لبيانات معينة من خلال Laptop أو Smart Phone موصول بشبكة الانترنت من خلال wifi. والبيانات التي تريد الوصول لها مثلاً عبارة عن مقطع Youtube مخزن في مركز بيانات شركة Youtube. لكن كيف تستطيع الوصول لذلك المقطع؟



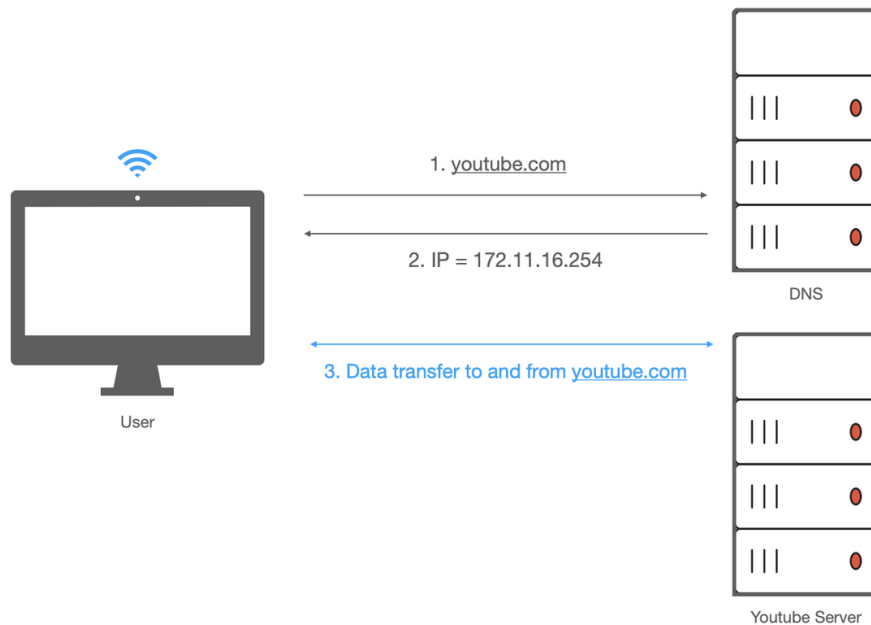
أي جهاز موصول بشبكة الانترنت يكون معرّف من خلال مجموعة من الأرقام تسمى IP address. تلك العناوين يتم توفيرها من خلال *ISP (Internet Service Provider)* والتي تقوم بتوفير عنوان مختلف لكل جهاز في شبكة الانترنت. كذلك Server الذي يحتوي على المقطع الذي تريد مشاهدته يكون له عنوان IP خاص به.



فمن خلال عناوين IP يكون هناك مُرسل Sender ومُستقبل Receiver.



وهذا يعني أن موقع Youtube يملك عنوان IP خاص به، لكن للبحث عنه والوصول إليه لا يشترط أن نقوم بحفظ عنوان IP. يمكننا استخدام الرابط `youtube.com` وذلك بمساعدة *DNS* الذي يقوم بربط كل عنوان IP باسم يسهل حفظه والبحث عنه باستمرار.

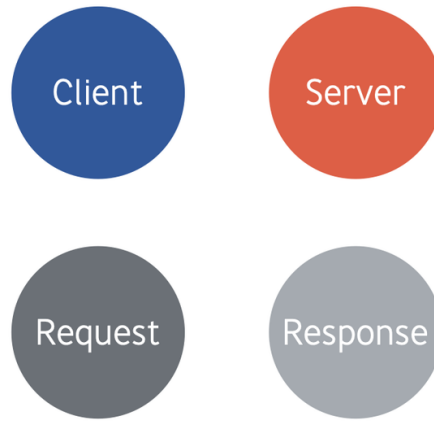


مكونات URL

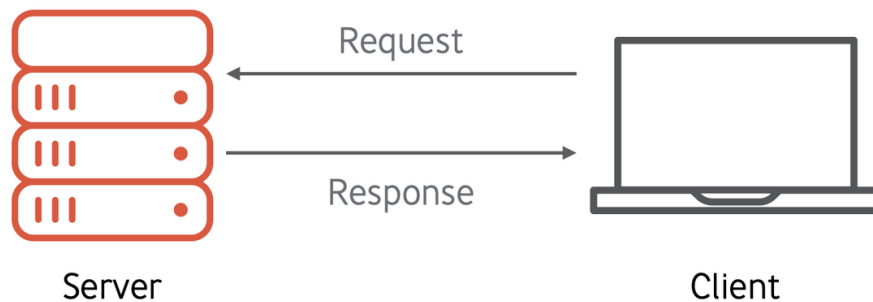


مفهوم World Wide Web

قبل البدء في تطوير مواقع الويب، نحتاج أولاً إلى معرفة مفهوم الويب بشكل عام. الويب هو عبارة عن نظام من صفحات الويب التي نستطيع الوصول إليها من خلال الإنترنت. مما يعني أن الويب ليس الإنترنت، بل هو واحد من التطبيقات التي تم إنشاؤها على الإنترنت. هناك بعض الأجزاء في World Wide Web نحتاج إلى فهمها ومعرفة الغرض من وجودها لكثرة استخدام مصطلحاتها أثناء تطوير تطبيقات الويب وهي Client, Server, Request, Response.



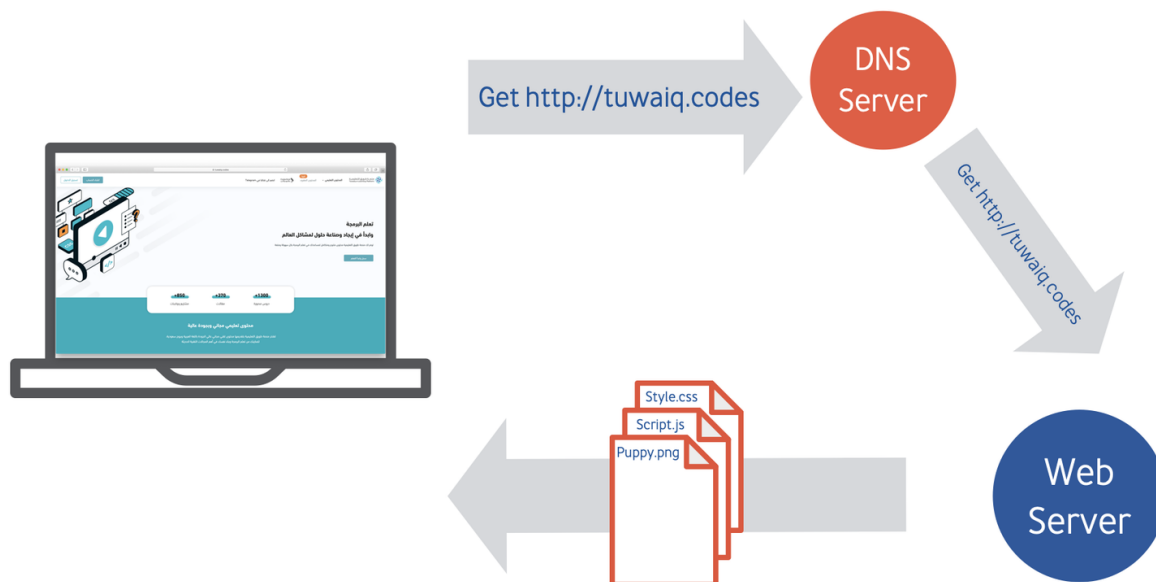
عندما يصفح المستخدم الإنترنت من خلال PC, Laptop, Smart Phone, iPad في هذه الحالة يعتبر **مستخدم (Client)**، بحيث أنه عندما يبحث عن موقع معين من خلال وضع رابط هذا الموقع في المتصفح فهو في هذه الحالة يقوم بإرسال **طلب (Request)** أي طلب لهذا الموقع، في هذه الأثناء يقوم **الخادم (Server)** باستقبال طلب المستخدم للموقع ثم يقوم بعرض هذا الموقع للمستخدم **كرد (Response)** لطلب المستخدم.



مثال:

عندما تقوم بفتح المتصفح (safari أو google chrome مثلاً) وتقوم بالبحث عن منصة طويق

التعليمية من خلال الرابط www.tuwaiq.codes في هذه الحالة أنت Client، ويقوم بعدها DNS باستقبال طلبك لتحويل ذلك الرابط إلى IP address الخاص بمنصة طويق التعليمية، حتى يرسل طلبك إلى Server منصة طويق التعليمية، بعد ذلك يقوم Server منصة طويق التعليمية بالرد على طلبك من خلال عرض الموقع لك.

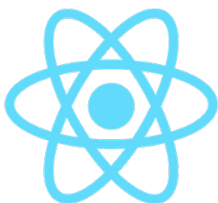


ما هو DNS؟ هو اختصار لمصطلح Domain Name System يستخدم لربط اسم الموقع أو رابط الموقع بعنوان IP address لذلك الموقع، لأنه من الصعب أن نقوم بحفظ IP address لكل المواقع التي نريد زيارتها، فيقوم DNS بالمساعدة بحيث يربط اسم كل موقع بعنوان الحقيقي للخادم Server الخاص به. فتبحث أنت باستخدام اسم الموقع مثل google.com ويقوم DNS بإيجاد العنوان المقابل 216.58.212.110 وإرسال Request للخادم Server الخاص بذلك الموقع.

Domain Name	IP Address
google.com	216.58.212.110
tuwaiq.codes	46.101.121.244

العناوين ليست دقيقة، مجرد مثال.

ماهي Web Frameworks؟



يوجد لدينا Web Frameworks مشهورة مثل: ReactJS، Angular، Vue، Django ويمكن تصنيف هذه Frameworks إلى ثلاثة أنواع كالتالي:

- **أولاً: Front-end Frameworks** وهي تعبر عن كل ما يراه المستخدم أو الأجزاء التي يتعامل معها المستخدم أثناء تصفح واجهات الموقع مثل الألوان والخطوط و dropdown menus وبشكل عام مكونات HTML و CSS و JavaScript.
- **ثانياً: Back-end Frameworks** وهي ما يمكن Front-end Frameworks من العمل ويقوم بحفظ البيانات الخاصة بالموقع وتشمل أيضا server و application.
- **ثالثاً: Full-stack Frameworks** وهي تشمل النوعين السابقين، حيث يعمل Full-stack Developer على Client-side و Server-side .

Front-end

1. How things look
2. Images, content, structure
3. HTML, CSS, JavaScript

Back-end

1. How things work
2. Logic & data
3. Ruby, Python, PHP, Java, etc



مفهوم MVC و MVT

مقدمة عن Django

يعتبر **Django** من أشهر Framework مفتوحة المصدر الخاصة بتطبيقات الويب ويسخدم في مواقع مشهورة مثل: Disqus, Instagram, Pinterest.



بالنسبة إلى **Django** فهو **Full-stack Framework** مسؤول عن بناء واجهات صفحات الويب عن طريق Templates ويتفاعل مع قواعد البيانات عن طريق Models. ويمكن أيضا أن نستخدم Django ك Back-end Framework و نحسن واجهات Django أو نضيف لها Interactivity عن طريق Front-end Frameworks مثل: ReactJS.

مميزات Django Framework:

- يعتبر batteries included web framework والمقصود هو أن Django تقدم العديد من المميزات مثل: libraries و packages التي تسهل عملية تطوير المواقع على المبرمج ومن أمثلة ذلك: ORM التي تساعد في بناء قواعد البيانات باستخدام لغة python دون الحاجة لكتابة أكواد بلغة SQL، أيضا توفر واجهات Admin جاهزة وغيرها.
- يساعد في إنشاء تطبيقات تتميز بأنها مرنة (reliable) بحيث يكون لديها القدرة في أنها تكبر لتناسب مع احتياجاتنا.
- يساعد في إنشاء تطبيقات آمنة (secure) وسهلة التعلم.
- لديها built-in database querying و URL mapping و template rendering.

المهارات التي نحتاجها لتعلم Django هي:

- معرفة أساسيات لغة Python
- معرفة أساسيات HTML, CSS, JavaScript
- معرفة أساسيات قواعد البيانات SQL



الأدوات التي نحتاجها لتعلم Django هي:

- تحميل Python نسخة 3 وأعلى.
- تحميل Editor مناسب مثل PyCharm.
- تحميل Django نسخة 3 وأعلى.



مفهوم المشروع والتطبيق في Django

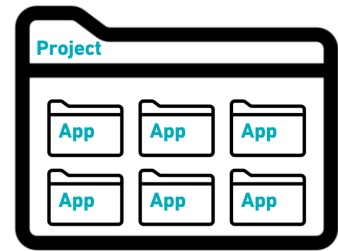
يمثل المشروع والتطبيق مفهومين مختلفين في Django حيث أن:

المشروع (project): يشير للخطوة التي نقوم فيها بتحميل Django مع الإعدادات الخاصة به.

التطبيق (application): يمثل مجموعة من models و views و templates و URLs.

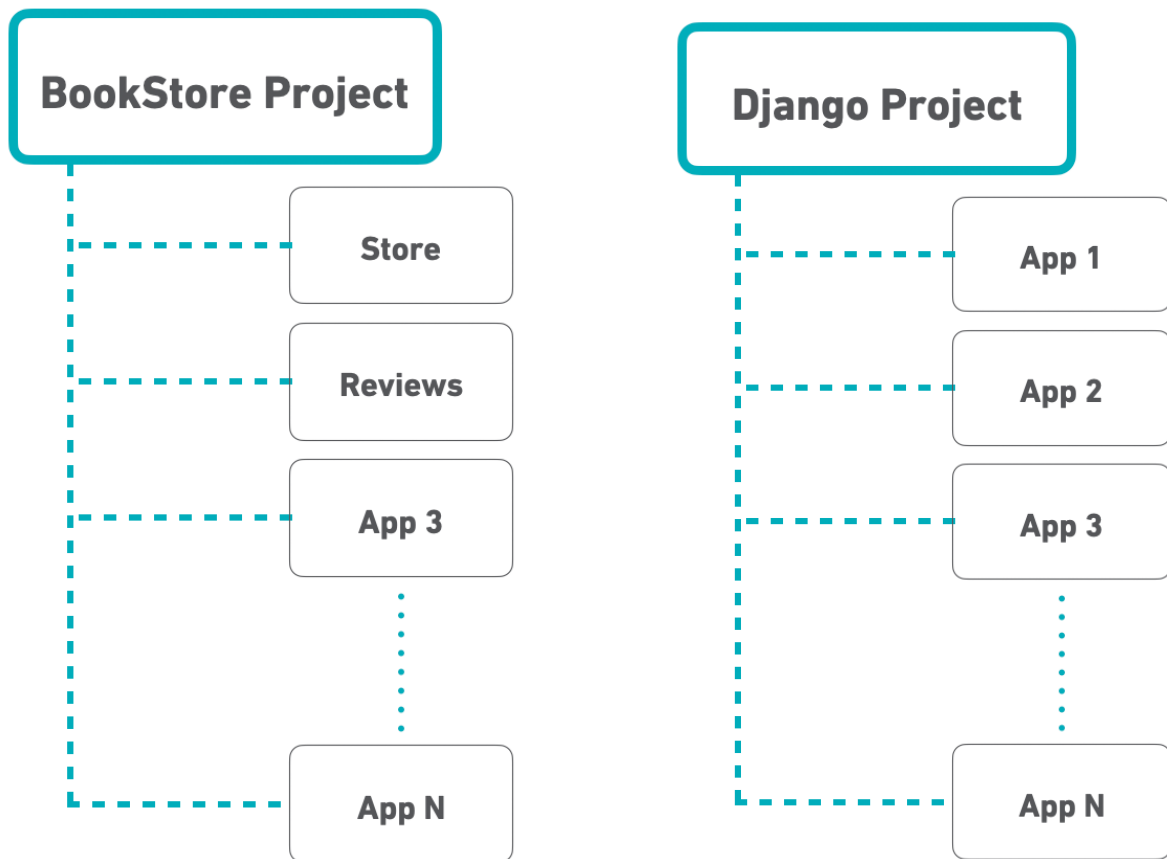
المشروع يمثل الموقع (website) والذي يتكون من عدة تطبيقات (applications) مثل: blog و forum ويمكن إعادة

استخدام التطبيقات في مشاريع أخرى.



مثال:

في Django نقسم المشروع لعدة تطبيقات على حسب الوظائف الخاصة فيها، لو كان لدينا مشروع bookstore يحتوي عدة تطبيقات مثل store app ، reviews app وغيره، نلاحظ أن كل تطبيق يتميز عن الآخر بالوظائف الخاصة التي يقدمها بحيث يمكن استخدام أحد هذه التطبيقات في مشروع آخر (مثال: استخدام تطبيق reviews في مشروع Movies).



إنشاء مشروع في Django

لإنشاء مشروع جديد في Django نقوم باتباع الخطوات التالية:

1. فتح برنامج المحرر.
2. اختيار New Project.
3. تسمية المشروع MoviesProject. نلاحظ أن PyCharm أنشأ بيئة افتراضية (Virtual Environment) للمشروع بشكل تلقائي وفائدة وجود بيئة افتراضية هي لحفظ ملفات Python packages للمشروع بشكل منفصل عن بقية (System Packages).
4. التأكد من اختيار أحدث نسخة Python.
5. فتح PyCharm Terminal وكتابة الأوامر التالية.

تحميل django بداخل Virtual Environment.

```
python -m pip install django
```

ملاحظة: أثناء تحميل Django قد يظهر لك التحذير التالي.

```
WARNING: You are using pip version xx.x.x; however, version xx.x.x is available.
```

عندها يمكن تحديث pip عن طريق كتابة:

```
python -m pip install --upgrade pip
```

إنشاء مشروع جديد بإسم Movies.

```
django-admin startproject Movies
```

سوف نلاحظ أن هذا الأمر أنشأ Python package بإسم المشروع Movies وملف manage.py
ملاحظة: إذا كانت الملفات غير ظاهرة لك قم بالضغط على الزر الأيمن واختيار Reload from the disk بعد ذلك سوف تظهر لنا الملفات.

لتشغيل السرفر ، نكتب الأمر التالي في الترمinal :

- 1 - `cd Movies`
- 2 - `python manage.py runserver`

الان، سوف يظهر لنا رابط (<http://127.0.0.1:8000> localhost) نقوم بالدخول عليه،
ونلاحظ ظهور العبارة the install worked successfully, congratulation وهذا يعني أن
المشروع يشتغل بشكل صحيح على server.
يمكن إيقاف server عن طريق الضغط على (Ctrl+C).

ملاحظة: أثناء تشغيل server قد يظهر لك التحذير التالي.

- 1 You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
- 2 Run 'python manage.py migrate' to apply them.

وهذا بسبب أن Django تنشأ تطبيقات افتراضية عند إنشاء المشروع مثل: تطبيق Admin و تطبيق
auth و sessions و contenttypes ولأن هذه التطبيقات تم إنشائها من دون عمل migration
يظهر لنا هذا التحذير لذلك يمكننا تجاهل التحذير أو تطبيق migration عن طريق الأمر.

```
python manage.py migrate
```

وسنتعرف على هذا الأمر بشكل مفصل عند العمل مع قواعد البيانات.