Task 2. Why Django is so popular among web developers?

One of the most significant advantages of Django is that it is simple to use. Developers can use it with a lot of ease. It's equipped with scalable and addons for developing the application that manages the extensive collection of data or high traffic. Additionally, it contains one of the best security systems among web development frameworks, which helps users avoid various security-related issues. Also, Django can be used to develop any project with ease. It is the perfect choice for the development of both straightforward and high-load applications. It helps to deal with major databases and assists in using them properly for projects. Developers can use multiple databases for their projects simultaneously. Overall, Django is one of the best popular framework among web developer.

Task 3. Five large companies that use Django

1. YouTube by Google
Originally it was a PHP-based project, but the constant need to improve its performance and add new functionalities forced YouTube to turn to Python as well.

2. Instagram by Meta
According to Statista, this site has 1 billion monthly active users. The Django framework helps Instagram perform all these tasks seamlessly.

3. Dropbox by Dropbox, Inc
Dropbox is a popular cloud storage service for pictures, videos, and documents. It's available online letting millions of users can access it regardless of their location. Dropbox uses the Django framework to facilitate synchronization, enable sharing options, and allow storage of large files.

4. Pinterest by Pinterest, Inc
Pinterest is a visual discovery engine that allows users to find ideas in their field of interest. For instance, fashion, recipes, healthcare, fitness, and home decor. It utilizes the Django framework to execute multiple tasks, scale, and maintain top-notch performance.

5. National Geographic by National Geographic Society
National Geographic used the Django framework to create a content management system (CMS) that handles all of the content on their website, including images and adverts.

Task 4. For each of the following scenarios, why I would use Django or wouldn't use :
- If I need to develop a web application with multiple users,
I would use Django because it's suitable for dealing with huge number of user's access.

- If I need fast deployment and the ability to make changes as you proceed,
I would use Django since it allows to fast developing.

- If I need to build a very basic application, which doesn't require any database access or file operations,

I wouldn't use Django because Its most notable feature is letting us to use multiple database and handle the extensive collection of data.

- If I want to build an application from scratch and want a lot of control over how it works,
I wouldn't use Django because I have to structure projects in a pre-specified way and do things in a certain way

- If I'm about to start working on a big project and are afraid of getting stuck and needing additional support,
I would use Django because of its huge community. It's easy to find out a solution provided by other developer.

```
(web-dev) sugasawanozomi@nojoling ~ % pip install django
Collecting django
  Downloading Django-4.2.13-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.6.0 (from django)
  Downloading asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Downloading sqlparse-0.5.0-py3-none-any.whl.metadata (3.9 kB)
Collecting backports.zoneinfo (from django)
  Downloading backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl.metadat
a (4.7 kB)
Collecting typing-extensions>=4 (from asgiref<4,>=3.6.0->django)
  Downloading typing_extensions-4.12.0-py3-none-any.whl.metadata (3.0 kB)
Downloading Django-4.2.13-py3-none-any.whl (8.0 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.0/8.0 MB 63.7 MB/s eta 0:00:00
Downloading asgiref-3.8.1-py3-none-any.whl (23 kB)
Downloading sqlparse-0.5.0-py3-none-any.whl (43 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 44.0/44.0 kB 9.0 MB/s eta 0:00:00
Downloading backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl (35 kB)
Downloading typing_extensions-4.12.0-py3-none-any.whl (37 kB)
Installing collected packages: typing-extensions, sqlparse, backports.zoneinfo,
asgiref, django
Successfully installed asgiref-3.8.1 backports.zoneinfo-0.2.1 django-4.2.13 sqlp
arse-0.5.0 typing-extensions-4.12.0
(web-dev) sugasawanozomi@nojoling ~ % django-admin --version
4.2.13
```

Download and install Python

```
sugasawanozomi@nojoling ~ % mkvirtualenv achievement2-practice
created virtual environment CPython3.8.7.final.0-64 in 204ms
  creator CPython3Posix(dest=/Users/sugasawanozomi/.virtualenvs/achievement2-pra
ctice, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle
, via=copy, app_data_dir=/Users/sugasawanozomi/Library/Application Support/virtu
alenv)
    added seed packages: pip==24.0, setuptools==69.5.1, wheel==0.43.0
  activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerS
hellActivator,PythonActivator
virtualenvwrapper.user_scripts creating /Users/sugasawanozomi/.virtualenvs/achie
vement2-practice/bin/predeactivate
virtualenvwrapper.user_scripts creating /Users/sugasawanozomi/.virtualenvs/achie
vement2-practice/bin/postdeactivate
virtualenvwrapper.user_scripts creating /Users/sugasawanozomi/.virtualenvs/achie
vement2-practice/bin/preactivate
virtualenvwrapper.user_scripts creating /Users/sugasawanozomi/.virtualenvs/achie
vement2-practice/bin/postactivate
virtualenvwrapper.user_scripts creating /Users/sugasawanozomi/.virtualenvs/achie
vement2-practice/bin/get_env_details
(achievement2-practice) sugasawanozomi@nojoling ~ % workon ahievement2-practice
ERROR: Environment 'ahievement2-practice' does not exist. Create it with 'mkvirt
ualenv ahievement2-practice'.
(achievement2-practice) sugasawanozomi@nojoling ~ % workon achievement2-practice

(achievement2-practice) sugasawanozomi@nojoling ~ %
```

Create a virtual environment: achievement2-practice

```
(achievement2-practice) sugasawanozomi@nojoling ~ % pip install django
Collecting django
  Using cached Django-4.2.13-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.6.0 (from django)
  Using cached asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.5.0-py3-none-any.whl.metadata (3.9 kB)
Collecting backports.zoneinfo (from django)
  Using cached backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl.metada
ta (4.7 kB)
Collecting typing-extensions>=4 (from asgiref<4,>=3.6.0->django)
  Using cached typing_extensions-4.12.0-py3-none-any.whl.metadata (3.0 kB)
Using cached Django-4.2.13-py3-none-any.whl (8.0 MB)
Using cached asgiref-3.8.1-py3-none-any.whl (23 kB)
Using cached sqlparse-0.5.0-py3-none-any.whl (43 kB)
Using cached backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl (35 kB)
Using cached typing_extensions-4.12.0-py3-none-any.whl (37 kB)
Installing collected packages: typing-extensions, sqlparse, backports.zoneinfo,
asgiref, django
Successfully installed asgiref-3.8.1 backports.zoneinfo-0.2.1 django-4.2.13 sqlp
arse-0.5.0 typing-extensions-4.12.0
(achievement2-practice) sugasawanozomi@nojoling ~ % django-admin --version
4.2.13
```

Install Django and verify the installation by checking the version