

1. Which of the following is the proper syntax to run the `migrate` command?

1 / 1 point

- ☐ `pip3 manage.py migrate`
- ☐ `migrate python manage.py`
- ☐ `python migrate manage.py`
- ☒ `python manage.py migrate`

✓ Correct

That's correct! The default port number is 3306, but you can adjust this is necessary.

2. Which of the following commands is used to enable or disable Django's transaction management?

1 / 1 point

- ☐ `CONN_MAX_AGE`
- ☒ `AUTOCOMMIT`
- ☐ `OPTIONS`
- ☐ `ENGINE`

✓ Correct

That's correct! When we set `AUTOCOMMIT` to false, it will disable Django's transaction management so we can implement our own.

3. Django places app-specific models in the _____ file in the app package folder.

1 / 1 point

- ☒ `models.py`
- ☐ `modelSettings.py`
- ☐ `py.models`
- ☐ `settings.py`

✓ Correct

That's correct! The `models.py` folder stores app-specific models.

4. True or False: It is mandatory to add the app to the installed apps list defined in the project `settings.py` module.

1 / 1 point

- ☒ True
- ☐ False

✓ Correct

That's correct! It is mandatory to add as this allows Django ORM to translate the model definitions into corresponding tables.

5. Which of the following statements are true regarding the differences between a Django API and a Django web application? Select all that apply.

0 / 1 point

- ☒ An API is a service used by any HTTP client, while a Django web application interacts mainly with a browser client.

✓ **Correct**

That's correct! Any HTTP client can interact with the API server, while a Django web application interacts mainly with a browser.

- ☐ The API needs user authentication, while the Django web application doesn't have the provision of authentication.
- ☒ An API returns a JSON or XML response to the client, while a Django web application usually returns an HTML response.

✓ **Correct**

That's correct! An API usually returns a JSON response, while a Django web application returns an HTML response.

6. Suppose we created a class which inherits both `RetrieveUpdateAPIView` and `DestroyAPIView` classes, which three HTTP methods will be supported?

0.8 / 1 point

- ☒ **PUT**

✓ **Correct**

That's correct! The `RetrieveUpdateAPIView` class supports the **PUT** method.

- ☒ **DELETE**

✓ **Correct**

That's correct! The `DestroyAPIView` class supports the **DELETE** method.

- ☐ **POST**

- ☒ **GET**

✓ **Correct**

That's correct! The `RetrieveUpdateAPIView` class supports the **GET** method.

7. True or False: When using DRF you can use the `DefaultRouter` class to wire up API URL's instead of declaring routes in the app and including them in the project's `URLConf`.

1 / 1 point

- ☒ True
☐ False

✓ **Correct**

That's correct! One of the things DRF makes easier is creating routes.

8. Which of the following parameters define the maximum lifetime of a connection?

1 / 1 point

- ☒ `CONN_MAX_AGE`
☐ `AUTOCOMMIT`
☐ `OPTIONS`
☐ `CONN_MAX_LIFE`

✓ **Correct**

That's correct! This parameter defaults to 0 which preserves historical behavior of closing the database connection at the end of each request.

9. What is one advantage of using `ViewSets` in Django REST Framework?

1 / 1 point

- ☐ Using one `ViewSet` we can handle all types of HTTP requests.
☐ It automatically enforces authentication.
☐ It automatically serializes the queryset.
☒ Views based on `ViewSets` are more concise and help to keep logic organized.

✓ **Correct**

That's correct! `ViewSets` make the logic compact and efficient.

10. This is a special type of serializer which quickly creates a serializer class from Django model fields.

1 / 1 point

- ☒ `ModelSerializer`
☐ `MenuSerializer`
☐ `JsonSerializer`

✓ **Correct**

That's correct! This serializer can save a significant amount of time when creating API's.