



REPLIQ Django Quiz

Total points 22/37 ?

The purpose of this simple quiz is to survey your overall understanding of Django

Email *

shekhrasel59@gmail.com

✗ When should one use serializers.ModelSerializerMethodField(...)? *

0/1

- ☐ When I want to represent some data in a model in a different way than the way it is stored in the model
- ☐ When I want to calculate a certain value not related to the model
- ☐ When I want to ignore model data but return something useful
- ☒ All of the above ✗
- ☐ None of the above

Correct answer

- ☒ When I want to represent some data in a model in a different way than the way it is stored in the model

✗ What is the purpose of Proxy models in Django? *

0/2

- ☐ Proxy models share the same table but You can customize its logic and behavior in Python
- ☐ Proxy models work as interfaces that every instance inherits to share the same logic
- ☐ You set proxy = True in the class Meta to use a Django proxy model
- ☒ All of the above ✗
- ☐ None of the above

Correct answer

- ☒ Proxy models share the same table but You can customize its logic and behavior in Python

✗ Is Django opinionated? *

0/1

- ☐ Yes!
- ☒ It depends, what do You mean by opinionated? ✗
- ☐ No, frameworks do not have opinions

Correct answer

- ☒ Yes!

✓ What is the purpose of managers in Django? *

2/2

- ☐ Managers can handle logic related to many rows in the database

- ☐ You can write custom methods to return filtered querysets
- ☐ You can write methods for creating new instances
- ☐ You can write methods to update many rows at once
- ☒ All of the above ✓
- ☐ None of the above

✗ How would You extend/customize Django's User model with custom fields? *0/2

- ☒ I would define a One-To-One field named "profile" and connect it to a model named Profile and then define my fields there ✗
- ☐ I would use the AbstractBaseUser class and then extend it with a CustomUser model
- ☐ I would write my own class from scratch in order to customize logic
- ☐ I would write a signal that connect the built in User model to a second model that will hold my custom fields
- ☐ All of the above makes sense
- ☐ None of the above makes sense

Correct answer

- ☒ I would use the AbstractBaseUser class and then extend it with a CustomUser model

✓ What does a normal serializer do Django REST Framework? * 1/1

- ☐ It stores session data
- ☐ It stores instance data
- ☒ It has logic for converting between JSON data and Python objects ✓
- ☐ It handles conversion logic between XML and HTML

✗ Is the Django ORM slow? * 0/2

- ☐ Yes
- ☐ Yes, it's at least slower than raw sql queries
- ☐ No
- ☒ It depends ✗
- ☐ All of the above
- ☐ None of the above
- ☐ Not if You know what it does

Correct answer

- ☒ Yes, it's at least slower than raw sql queries

✓ How would You handle static files in Django? * 2/2

- ☐ I would upload them via an FTP call to a FTP server and then store the path in the database
- ☒ I would store my files on AWS S3 or something similar and store the path in the database ✓
- ☐ I would store the whole file in my database as a BLOB

- ☐ I would setup a static files server on my laptop and run Apache on it to serve the files
- ☐ I would use a CDN service
- ☐ All of the above
- ☐ None of the above

✓ What does the root URL file do in Django? * 1/1

- ☐ It enlists all the REST endpoints
- ☒ It is like an index of all the urls of a Django project ✓
- ☐ It holds all the urls in a certain module

When should You use Model.objects.select_related(...)? *

- ☒ When I want to fetch known field data from related tables and not fetch the data lazily
- ☐ When I want to load many rows at once and cache the queryset
- ☐ When I want to load relational meta data before I traverse the data in Python

✓ What is the purpose of Abstract classes? * 2/2

- ☐ An abstract class is a class that uses the same table but You can specify customized behavior
- ☒ An abstract class is a blueprint that other classes can use in order to inherit certain attributes and methods ✓
- ☐ An abstract class is a class which defines an interface so that other classes can expect a defined set of behaviours from it

✓ What does MVT stand for? * 1/1

- ☐ Model View Tag
- ☐ Model Validation Tool
- ☒ Model View Template ✓
- ☐ Model Verification Tag

✓ What type of logic should You store in a model? * 3/3

- ☐ Database validation logic
- ☐ Logic related to an instance/row in the database
- ☐ Store column information and meta data
- ☐ Primary keys and their relationships
- ☐ Store index related configuration
- ☒ All of the above ✓
- ☐ None of the above

✗ What is a template in Django? * 0/2

- ☒ Templates are usually text files that are placeholders for static and dynamic content ✗
- ☐ Templates are files that contain HTML
- ☐ Templates are files that contain XML
- ☐ All of the above
- ☐ None of the above

Correct answer

- ☒ All of the above

✗ How would You write a Django REST Framework view that only allows for ^{*0/3} changing a single field in a model thru a REST call?

- ☐ Create a serializer with one defined field and map it to a UpdateView view where url can be something like `'/api/v1/fruits/<slug>/fieldname'`
- ☒ Use a ModelSerializer and expose one field and then make it accessible so that the developer can make a PATCH request ✗
- ☐ Define a raw function that takes the form data and manually validate it and then sets a specific field
- ☐ I would use a ListView with a custom serializer that checks the request data and removes all other data except the data related to one field
- ☐ I would define a Django ModelForm instance with one single exposed field

Correct answer

- ☒ Create a serializer with one defined field and map it to a UpdateView view where url can be something like `'/api/v1/fruits/<slug>/fieldname'`

✓ What is REST? *

1/1

- ☐ A protocol for storing data in the backend
- ☐ A client that handles communication between the server and the database
- ☒ A method to communicate between client and server ✓
- ☐ An HTTP layer that handles communication between two parties
- ☐ All of the above
- ☐ None of the above

✓ Can You store a Django model in an ElasticSearch index? *

2/2

- ☐ Yes, if You serialize the data and store it as XML
- ☒ Yes, if You serialize the data and store it as JSON ✓
- ☐ Yes, only if You strip the HTML tags
- ☐ Yes, as a string
- ☐ No

✗ What does the Django ORM do? *

0/1

- ☐ It handles all interaction with the database
- ☐ It maps relational objects from the database into native Python objects
- ☐ It handles the sessions with the database
- ☐ It handles the connections with the database

☒ All of the above ✗

☐ None of the above

Correct answer

☒ It maps relational objects from the database into native Python objects

✗ You want to store a floating point number in a model. What built-in model field would You use to store it? *0/1

☐ DecimalField

☐ IntegerField

☐ CharField

☒ FloatField ✗

☐ All of the above

☐ None of the above

Correct answer

☒ DecimalField

✓ Can You handle REST calls with Django out of the box? * 1/1

☒ Yes ✓

☐ No

☐ Maybe

✓ What is the purpose of MVC frameworks? * 2/2

☐ Clear separation of logic

☐ More readable code

☐ Better project structure

☒ All of the above ✓

☐ None of the above

✓ Does Django handle migrations for You? * 1/1

☒ Yes ✓

☐ Only if You write some specific logic

☐ No

✓ What is Celery in the context of Django? * 1/1

☐ It is a vegetable that is used by the community as a mascot

☐ It is cache invalidation service customized for Django

☒ It is an asynchronous task queue that plays well with Django ✓

☐ All of the above

☐ None of the above

✓ What database engine would You preferably use with Django? *

1/1

- ☐ MySQL
- ☒ PostgreSQL
- ☐ MongoDB
- ☐ SQLite
- ☐ Redis
- ☐ ElasticSearch
- ☐ Memcached
- ☐ Celery
- ☐ GraphQL



✓ What is a Django management script? *

1/1

- ☒ It is a way to run code outside of the request/response loop
- ☐ It is a script that generates static HTML in a Django project
- ☐ It is some arbitrary code that runs every hour to do some cleanup
- ☐ All of the above
- ☐ None of the above



This form was created inside of HealthOS AB. [Report Abuse](#)

Google Forms

