

In the lab you just completed, you created two models, **Drinks** and **DrinkCategory**, and registered them using the function `admin.site.register()`. Which of the following statements about model registration using `admin.site.register()` is true?

1 / 1 point

- ☐ Model registration using `admin.site.register()` must be performed before you can edit the model using the Django shell.
- ☐ Model registration using `admin.site.register()` must follow the order in which the classes are defined in the `models.py` file?
- ☒ Model registration using `admin.site.register()` allows the model to be accessed from the admin user interface.

✓ Correct

Correct! Model registration using `admin.site.register()` is required to access the model from the admin user interface.

2. In the lab you just completed, you defined a many-to-one relationship using `django.db.models.ForeignKey`. One of the arguments passed to this field type was `on_delete = 'models.PROTECT'`. What is the purpose of this argument?

1 / 1 point

- ☐ The model object is deleted with a cascade effect that deletes all other referenced models.
- ☒ The **PROTECT** argument prevents the deletion of the model object that is referenced.
- ☐ The model object is deleted and the **ForeignKey** field type resets to a default value.

✓ Correct

Correct! The **PROTECT** argument prevents the deletion of the referenced object if an object references it in the database. For example you cannot delete the 'DrinkCategory' model if the 'Drink' model contains data that references drink categories in the associated database table.

To delete it, you must delete all objects that reference it manually.

3. In the lab you just completed, you created two models, **Drinks** and **DrinkCategory**. You defined a many-to-one relationship using `django.db.models.ForeignKey`.

Inside the `admin.py` file, which of the following code blocks are required to provide complete visibility of both models to the Admin panel? Select all that apply.

✓ ☒ 1 `from .models import Drink`

✓ Correct

Correct! The base model needs to be imported to ensure its visibility inside the Admin panel.

✓ ☒ 1 `from .models import DrinkCategory`

✓ Correct

Correct! The model object importing the **ForeignKey** of another model needs to be imported separately to ensure visibility.

✓ ☒ 1 `from django.contrib import admin`

✓ Correct

Correct! Importing `admin` is required for using the function `admin.site.register()` inside it

☐ 1 `from django.db import models`