



Lab 2: Build Course Exam Related Models and Templates

Estimated time needed: 80 minutes

In this lab, you will be creating the exam related models and templates to add a new exam section in course details page.

Objectives

This lab contains the following tasks:

1. Create Question, Choice, and Submission models
2. Create a new course object with exam related models using the admin site
3. Update the course details template to show questions and choices

Environment setup

If your Theia workspace has been reset and you want to continue from what you have done previously, please `gitclone` or pull from your created GitHub repository.

- Set up the Python runtime if Theia workspace has been reset.

```
pip3 install -r requirements.txt
```

Note that you may need to perform models migrations for a new Theia environment.

Note that the provided database is an empty database for the final project, if you want to see some course data on the course page, you could use admin site to create some new courses after migration.

Task: To build the new assessment feature related models

You will need to create several new models in `onlinecourse/models.py`:

- A `Question` model to save the questions of an exam
- A `Choice` model to save all choices of a question

You can find more hints in the code comments with tag `<HINT>`.

- Create a question class `class Question(models.Model):` Model:
 - Has a One-To-Many (or Many-To-Many if you want to reuse questions) relationship with Course model
 - The question text content
 - A grade point for the question
 - Other fields and methods you would like to include

Note for `Question` model, we provided a sample model method `is_get_score` (commented out) to evaluate if the question was answered correctly by comparing the selected choice ids with correct choices in the question.

You can use the sample `is_get_score` method directly or feel free to implement a similar one by yourself.

- Create a choice `class Choice(models.Model):` Model:

- One-To-Many (or Many-To-Many if you want to reuse choices) relationship with Question model
- The choice text content
- Indicate if this choice is a correct one or not
- Other fields and methods you would like to include

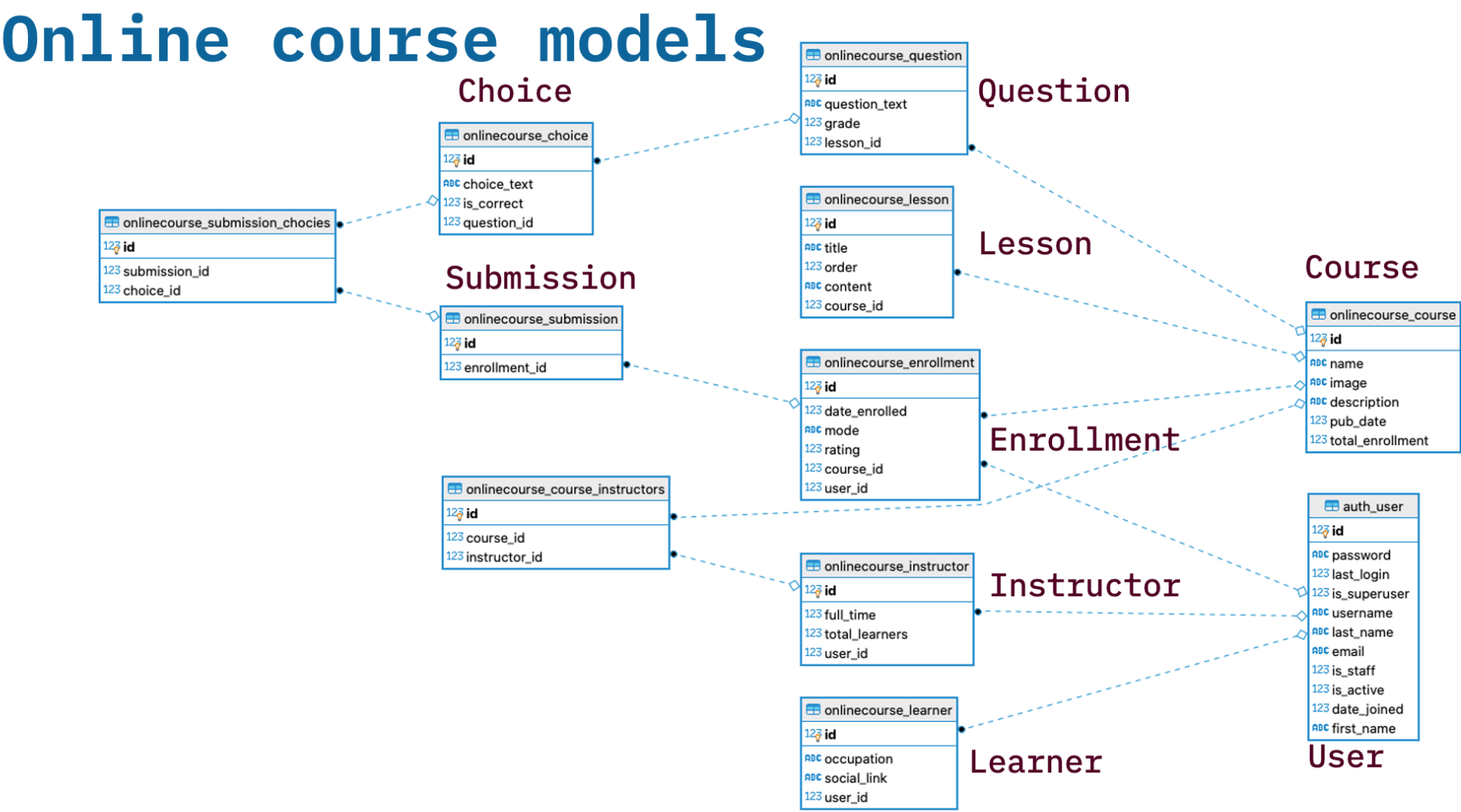
We provided a `Submission` model (commented out) which has:

- One-to-Many relationship with Enrollment, i.e., one course enrollment could have multiple exam submissions
- Many-to-Many relationship with choices or questions. For simplicity, you could just relate the submission with the Choice model

You need to uncomment the Submission model and use it to associate selected choices.

Refer to the previous ORM labs or other models in `models.py` as examples.

Here is an example ER Diagram for your reference:



- Run migrations for the models

```
python3 manage.py makemigrations onlinecourse
```

and

```
python3 manage.py migrate
```

Note if you see any errors related to model migrations, you could choose to delete the existing database `db.sqlite3` and re-run above migration again.

Task: To register Question and Choice models with the admin site.

- You need to register the `Question` and `Choice` in the admin site so that you could conveniently

manage their content.

- You will also need to create `QuestionInline` and `ChoiceInline` classes so that you could edit them

together on one page in the admin site.

- You need to create a super user for the admin site.

After you registered the new models, you could create a new course with lessons, questions, and question choices using the admin site.

Refer to the previous admin site lab or other models in `admin.py` as examples.

- Run Django development server and check if you are able to add

Question and Choice objects using admin site.

Task: To update the course detail template to create an exam section with a list of questions and choices

One exam contains multiple questions and each question should have more than one correct choice (multi-selection).

Start Exam

What is Django?

☒ A Web framework

☐ A Movie

What is Django Model

☒ The single, definitive source of information about your data

☐ A web framework

☐ Perform ORM for developers

What is Django View

☒ Class-based view

☒ Function-based view

☐ It is a Controller

Submit

- Update the template for course detail page in `../templates/onlinecourse/course_details_bootstrap.html`:
 - If user is authenticated, show course exam with a list of questions and choices
 - You may use `Bootstrap Collapse` to hide exam. We have provided a `collapse` example in comment.
 - Each example will have many questions
 - Each question will have many choices
 - For each question choice, you may create a checkbox input to save the selection
 - Create a form to collect the selected choice ids for all questions. We have provided a `form` example in comment.

Refer to the previous template and Bootstrap labs or other templates as examples.

- Run Django development server and check if you see an example section in course detail page.

At this moment, you can not submit the exam. You will be implementing that in the next lab.

Commit your updated project to GitHub

Commit all updates to the GitHub repository you created so that you could save your work.

If you need to refresh your memory on how to commit and push to GitHub in Theia lab environment, please refer to this lab [Working with git in the Theia lab environment](#)

Author(s)

[Yan Luo]([linkedin.com/in/yan-luo-96288783](https://www.linkedin.com/in/yan-luo-96288783))

Changelog

Date	Version	Changed by	Change Description
04-Dec-2020	1.0	Yan Luo	Initial version created

