Home Task for the
World of Warships: Legends

Position: Automation QA Engineer

Timeframe for completion: 7 days

Not restricted by NDA

Hello!

We truly appreciate your willingness to take on our Home Task.

At Wargaming, we believe that speed and high-quality work are our key advantages. That's why we are looking for people who can deliver strong results within a reasonable timeframe. Just like in real production.

Please note: we do not use the results of the Home Task anywhere beyond the evaluation process.

If you have any questions, please feel free to reach out to your Recruiter.

Good luck!

Home Task goal

The Home Task is designed to assess the work with Python, the level of the programming culture and knowledge. Task will show the candidate what he will have to do in our team. Despite this, the data in the task has a very simplified schema relative to the real one.

Requirements for the completed Task

- Python version – 3.10 or higher
- Tests must be written using the **pytest** framework
- Use pytest.mark.parametrize or the pytest_generate_tests hook for parameterization
- Should be **600** tests in total after run
- As a result of the assignment, there must be at least the following:
  - Script that creates and populates the source database
  - Python module with tests
  - conftest.py with fixtures/hooks (optional)
- Code style – PEP8

I TASK

Write a python script that creates a SQLite database according to the specified scheme.
Primary keys – weapon / ship / hull / engine are string fields.
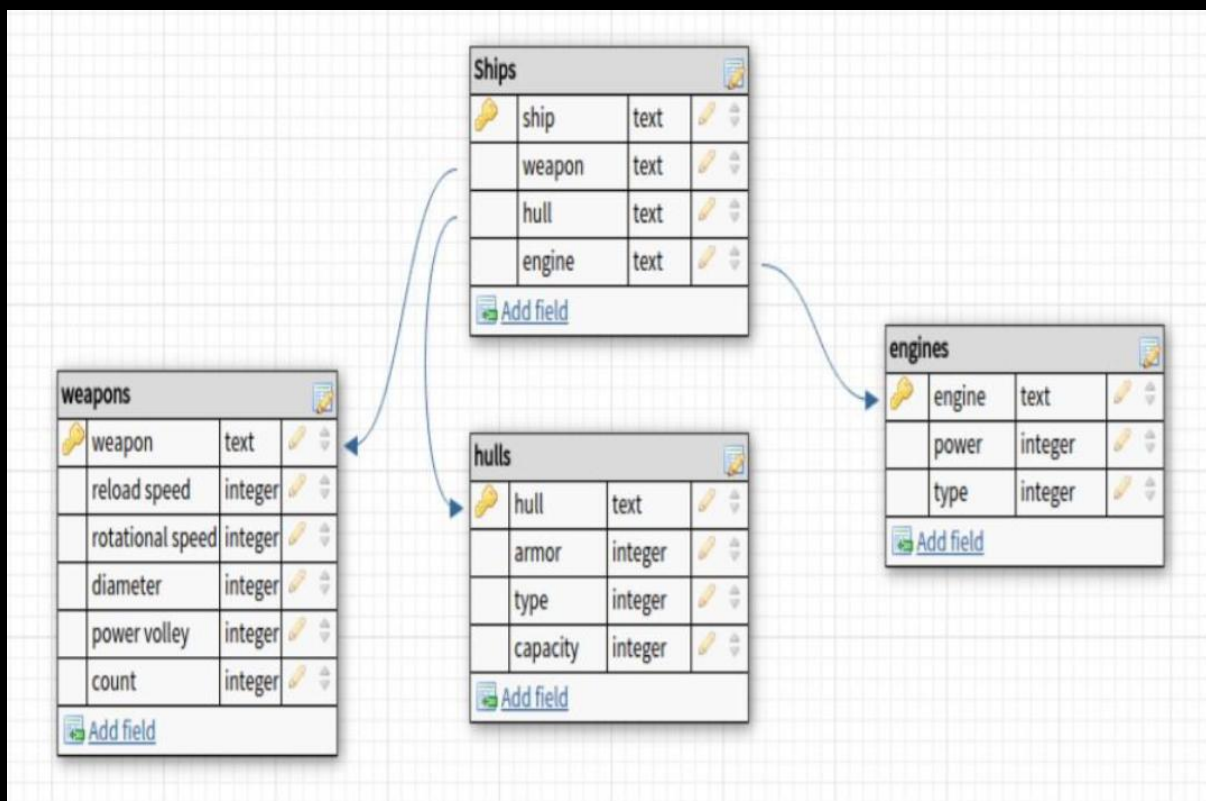
II TASK

Create a script that will randomly fill in the values in the created database. The names will fit perfectly: Ship-1, Ship-2, Weapon-1, etc.

Number of records for each table:

ships: **200**

weapons: **20**

hulls: **5**

engines: **6**

Value range for integer parameters: **1-20**

III TASK

Create a session-scope fixture that gets the current state of the database and creates a temporary new database where the values are randomized:

A. For each ship, it randomly changes one of the components: hull, gun or engine

B. In each component, one of the randomly selected parameters is changed to a random value from the allowable range (see above)

IV TASK

Create autotests that compare data from the original database with the randomized one:

    A. Each ship should have 3 tests that check its gun, hull and engine.

    B. Test should be failed:

       1. If the value of a component parameter does not match what it was before the randomizer was run.

       Output example:

```
Ship-2, weapon-1
    reload speed: expected 1, was 2
Ship-3, hull-3
    type: expected 1, was 2
Ship-3, engine-6
    power: expected 22, was 13
```

       2. If the gun, hull, or engine of the ship has changed.

       Output example:

```
Ship-5, engine-4
    expected engine-1, was engine-4
```