djoseph-png / **py-cinema-visit**

Code Pull requests Actions Projects Wiki Security Insights Settings

0 stars 1.8k forks 0 watching Branches Activity
Tags

Public repository · Forked from mate-academy/py-cinema-visit

1 Branch 0 Tags Go to file t Go to file Add file Code

This branch is up to date with `mate-academy/py-cinema-visit:master` . Contribute Sync fork

marina-tilniak Merge pull request mate-academy#1457 from pavlosoletskyy/patch-1

48b6d8e · 7 months ago

.github/workflows	Update test.yml	3 years ago
app	Init	3 years ago
tests	Add named args	3 years ago
.flake8	Update .flake8	3 years ago
.gitignore	Initial commit	3 years ago
README.md	Update README.md	11 months ago
checklist.md	fixes	11 months ago
requirements.txt	Update requirements.txt	3 years ago

README

Cinema visit

- Read the guideline before start

You have opened your own cinema. To have a better idea of what's going on in the cinema you decided to keep a record of events in the cinema. For this purpose you have to create such modules:

1. In directory `app` create package `cinema` . In this package create modules:
    - `bar.py` - inside this module create `CinemaBar` class that describes work of cinema bar. This class should have only one static method `sell_product` , that takes `product` - name of the product that customer wants and `customer` - `Customer` instance, that means customer. The `sell_product` method sells a product to the customer and displays which product was sold and to whom.

```
cb = CinemaBar()
customer = Customer("Bob", "popcorn")
cb.sell_product(customer=customer, product=customer.food)
# Cinema bar sold popcorn to Bob.
```

- `hall.py` - inside this module create `CinemaHall` class that describes actions during the movie session. Its `__init__` method takes and stores ONLY the `number` of the hall in the cinema. This class should have only one method `movie_session`, that takes `movie_name`, `customers` - list of a customers (`Customer` instances), `cleaning_staff` - cleaner (`Cleaner` instance). This method prints about movie start, calls customers method `watch_movie`, prints about movie end, calls cleaner method `clean_hall`. So, we are expecting that everything listed above will be performed in `movie_session` function.

```
hall = CinemaHall(hall_number=5)
movie_name = "Madagascar"
customers = [
    Customer(name="Bob", food="Coca-cola"),
    Customer(name="Alex", food="popcorn")
]
cleaning_staff = Cleaner(name="Anna")

hall.movie_session(movie_name=movie_name, customers=customers, cleaning_staff=cleaning_staff)
```

2. In directory `app` create package `people`. In this package create modules:

- `customer.py` - inside this module create `Customer` class, its `__init__` method takes and stores `name`, `food` - food that customer wants to buy in cinema bar. This class should have only one method `watch_movie`, this method takes `movie` and prints what movie customer is watching.

```
bob = Customer(name="Bob", food="popcorn")
bob.watch_movie(movie="Madagascar")
# Bob is watching "Madagascar".
```

- `cinema_staff.py` - inside this module create `Cleaner` class, its `__init__` method takes and stores `name`. This class should have only one method `clean_hall`, this method takes `hall_number` - number of hall that cleaner have to clean and prints that cleaner is cleaning that hall.

```
anna = Cleaner(name="Anna")
anna.clean_hall(hall_number=5)
# Cleaner Anna is cleaning hall number 5.
```

In the module `main.py` you have to import all this classes. Classes should be imported by absolute path, that starts with 'app.' with keyword 'from'. Write a function `cinema_visit` that takes `movie`, `customers` - a list of customers, elements are dicts with 'name' and desired 'food' of a customer, `hall_number` - number of the hall in cinema, `cleaner` - name of the cleaner, that will clean the hall after movie session.

This function should create instances of `Customer`, `CinemaHall`, and `Cleaner`. First, the cinema bar should sell food to customers. To do this, you can use the `CinemaBar` class without creating an instance. Then, the cinema hall should schedule a movie session, and finally, a cleaner should clean the cinema hall. We expect each class to work with the provided data, accepting parameters in the correct order and having the necessary methods. No additional checks or error handling are needed!

Example (do not add it to `main.py`):

```
customers = [
    {"name": "Bob", "food": "Coca-cola"},
    {"name": "Alex", "food": "popcorn"}
]
hall_number = 5
cleaner_name = "Anna"
movie = "Madagascar"
cinema_visit(customers=customers, hall_number=hall_number, cleaner=cleaner_name, movie=movie)
# Cinema bar sold Coca-cola to Bob.
# Cinema bar sold popcorn to Alex.
# "Madagascar" started in hall number 5.
# Bob is watching "Madagascar".
# Alex is watching "Madagascar".
# "Madagascar" ended.
# Cleaner Anna is cleaning hall number 5.
```

Important Note: Each method responsible for performing a task should only print a message using the `print()` function. There is no need to return anything or use the `logging` module.

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

- **Python** 100.0%

Suggested workflows

Based on your tech stack

Python package

Create and test a Python package on multiple Python versions.

Configure

Django

Build and Test a Django Project

Configure

Python application

Create and test a Python application.

Configure

More workflows

Dismiss suggestions