

Szoftver tesztelés és validáció projekt dokumentáció

Készítette:

Kaffai Levente, Számítástechnika 4.év, 2025

Tartalom

Szoftver tesztelés és validáció projekt dokumentáció	1
Bevezetés	3
Projekt struktúra	3
Teszt sor Struktúra	3
Fixtúrák	3
EmployeeManager osztály teszt esetei	3
Alapvető Fizetésszámítás Tesztelése	3
Vezetői Fizetésszámítás Tesztelése	3
Újonnan Felvett Alkalmazott Fizetésszámítása	4
Fizetésszámítás és E-mail Értesítés Tesztelése	4
Osztályváltozók Ellenőrzése	4
Vezető Fizetésszámítása Csapattagok Nélkül	4
Vezető Fizetésszámítása és E-mail Értesítés	4
RelationsManager osztály teszt esetei	4
Tesztelési Cél	4
Teszt esetek	5
Kódbázis teszt lefedettsége	5
CI/CD	6
Következtetés	6

Bevezetés

Ez a dokumentáció az Employee Manager osztály tesztjeit fogja össze, különös tekintettel az **employee_manager_test.py** és **relations_manager_test.py** modulokra. A tesztek biztosítják az EmployeeManager és RelationsManager osztályok helyességét és megbízhatóságát.

A futtatást Python 3.10.+ verzióban lett megvalósítva, egy python virtuális környezettel, ennek a környezetnek a követelményei a requirements.txt fájl tartalmazza.

Projekt struktúra

A projektben minden állomány amely tartalmazza a **_test** kulcsszót a nevében, egységtesztet tartalmaz, és futtatásra kerülnek a Pytest keretrendszerben, jelen dokumentáció főleg azokra fókuszál.

Teszt sor Struktúra

A teszt sor a Pytest keretrendszerrel van megszervezve. Tartalmaz fixtúrákat a teszt adatok beállításához és több teszt esetet különböző forgatókönyvek validálásához.

Fixtúrák

A fixtúrák olyan segéd funkciók, amelyek a tesztek során használt adatokat és objektumokat állítják be. A Pytestben a **@pytest.fixture** dekorátorral jelölik őket. A fixtúrák lehetővé teszik, hogy a tesztek között megosszunk kódot, így csökkentve a redundanciát és javítva a tesztek olvashatóságát.

- **relations_manager_mock**: A RelationsManager osztályhoz tartozó mock objektum, amely segít szabályozni a viselkedését a tesztek során.
- **employee**: Egy minta Employee objektum előre meghatározott attribútumokkal.
- **employee_manager**: Az EmployeeManager osztály példánya, amely a relations_manager_mock-ot használja.

EmployeeManager osztály teszt esetei

Alapvető Fizetésszámítás Tesztelése

test_calculate_salary_basic: Teszteli az alapvető fizetésszámítást egy nem vezető alkalmazott esetében.

- **Beállítás**: A **is_leader** metódust úgy konfigurálja, hogy False értéket adjon vissza.
- **Ellenőrzés**: Meggyőződik arról, hogy a kiszámított fizetés megfelel a vártnak, figyelembe véve a bázisfizetést és a szolgálati időt.

Vezetői Fizetésszámítás Tesztelése

test_calculate_salary_for_leader: Teszteli a fizetésszámítást egy vezető alkalmazott esetében, akinek csapattagjai vannak.

- **Beállítás**: A **is_leader** metódust úgy konfigurálja, hogy True értéket adjon vissza, és beállítja a csapattagokat.

- **Ellenőrzés:** Meggyőződik arról, hogy a kiszámított fizetés tartalmazza a vezetői bónuszokat a csapattagok után.

Újonnan Felvett Alkalmazott Fizetésszámítása

test_calculate_salary_for_new_employee: Teszteli a fizetésszámítást egy újonnan felvett alkalmazott esetében.

- **Beállítás:** Létrehoz egy olyan alkalmazottat, aki az adott évben kezdett el dolgozni.
- **Ellenőrzés:** Meggyőződik arról, hogy a fizetés a bázisfizetés, éves bónuszok nélkül.

Fizetésszámítás és E-mail Értesítés Tesztelése

test_calculate_salary_and_send_email: Teszteli a fizetésszámítást és az e-mail értesítési funkciókat.

- **Beállítás:** Mockolja a print függvényt az e-mail üzenet ellenőrzéséhez.
- **Ellenőrzés:** Meggyőződik arról, hogy a megfelelő fizetés kiszámításra kerül és az e-mail üzenet kiírásra kerül.

Osztályváltozók Ellenőrzése

test_class_variables: Ellenőrzi az EmployeeManager osztály változóit.

- **Ellenőrzés:** Meggyőződik arról, hogy a yearly_bonus és leader_bonus_per_member változók a vártnak megfelelő értékeket tartalmazzák.

Vezető Fizetésszámítása Csapattagok Nélkül

test_calculate_salary_for_leader_without_team: Teszteli a fizetésszámítást egy olyan vezető esetében, akinek nincsenek csapattagjai.

- **Beállítás:** Az is_leader metódust úgy konfigurálja, hogy True értéket adjon vissza, de nincsenek csapattagok.
- **Ellenőrzés:** Meggyőződik arról, hogy a fizetés helyesen van kiszámítva csapattagi bónuszok nélkül.

Vezető Fizetésszámítása és E-mail Értesítés

test_calculate_salary_and_send_email_for_leader: Teszteli a fizetésszámítást és az e-mail értesítést egy vezető esetében.

- **Beállítás:** Beállítja a csapattagokat és ellenőrzi az e-mail üzenetet.
- **Ellenőrzés:** Meggyőződik arról, hogy a megfelelő fizetés kiszámításra kerül és az e-mail üzenet kiírásra kerül.

RelationsManager osztály tesztesei

A RelationsManager osztályhoz tartozó tesztek a **relations_manager_test.py** állományban találhatók.

Tesztelési Cél

A RelationsManager osztály tesztelése során az alábbi funkciókat kell ellenőrizni:

- **test_is_leader:** Ellenőrzi, hogy egy alkalmazottat helyesen azonosítja-e vezetőként.

- **test_get_all_employees:** Teszteli az összes alkalmazott lekérdezését.
- **test_get_team_members:** Validálja a csapattagok lekérdezését egy vezető esetében és a nem vezetők kezelését.

Tesztesetek

test_is_leader

Leírás: Teszteli, hogy a RelationsManager helyesen azonosítja-e egy alkalmazottat vezetőként.

- **Beállítás:** Létrehoz egy vezető és egy nem vezető alkalmazottat.
- **Ellenőrzés:** Meggyőződik arról, hogy a is_leader metódus a vártnak megfelelő értékeket adja vissza.

test_get_all_employees

Leírás: Teszteli az összes alkalmazott lekérdezését.

- **Beállítás:** Létrehoz több alkalmazottat.
- **Ellenőrzés:** Ellenőrzi, hogy a get_all_employees metódus visszaadja-e az összes alkalmazottat.

test_get_team_members

Leírás: Teszteli a csapattagok lekérdezését egy vezető esetében.

- **Beállítás:** Létrehoz egy vezetőt és csapattagokat.
- **Ellenőrzés:** Meggyőződik arról, hogy a get_team_members metódus visszaadja-e a vezető csapattagjait.

Kódbázis teszt lefedettsége

----- coverage: platform win32, python 3.10.11-final-0 -----			
Name	Stmts	Miss	Cover

calculator.py	6	0	100%
calculator_test.py	10	0	100%
employee.py	10	0	100%
employee_manager.py	29	1	97%
employee_manager_test.py	84	0	100%
relations_manager.py	15	0	100%
relations_manager_test.py	24	0	100%

TOTAL	178	1	99%

Annak érdekében hogy a **employee_manager** modul magasabb teszt lefedettséggel rendelkezzen, a main metódus is tesztelésre került, ahol ellenőrzésre került, hogy a print metódus meghívásra került legalább egyszer.

CI/CD

Tesztek automatikus futtatása Github Action-el

A YAML fájl hozzáadása a projektbe általában a konfigurációs fájlok kezelésére szolgál. Egy GitHub Actions workflow esetében a YAML fájl tartalmazza a futtatandó lépéseket és a környezeti beállításokat. A GitHub Actions egy automatizált folyamatkezelő rendszer, amely lehetővé teszi, hogy a kód módosításaira válaszul automatikusan futtasson parancsokat vagy teszteket.

Hogyan működik a GitHub Actions?

1. **Workflow Fájl:** A YAML fájl a `.github/workflows` mappában található, és meghatározza a workflow lépéseit.
2. **Trigger:** A workflow egy adott eseményre indul el, például kód feltöltésekor (push) vagy pull request létrehozásakor.
3. **Futtatás:** A GitHub Actions a workflow lépéseit végrehajtja egy virtuális környezetben, például tesztek futtatása, kód összeállítása vagy telepítés.
4. **Logolás:** Az eredmények és hibák a GitHubon megtekinthetők.

Következtetés

A tesztsor átfogóan fedi le a különböző forgatókönyveket az alkalmazotti fizetések számításához és a csapatkezeléshez, biztosítva az EmployeeManager és RelationManager osztályok megbízhatóságát. A Github Actions funkcióival könnyen integrálható a tesztek automatikus futtatása virtuális környezetbe, ezáltal jelentősen megkönnyítve a tesztek helyességének ellenőrzését.