

מערכת הרשמה לקורסים - קובץ README

מבוא

מערכת ניהול קורסים אקדמיים זו מיועדת לשימוש במוסדות לימוד ומטרתה לאפשר ניהול יעיל של קורסים, רישום סטודנטים וניהול משתמשים. המערכת כוללת יכולות שונות כמו רישום והסרת רישום לקורסים, ניהול פרטי קורסים ועוד.

פעולות עיקריות בקוד

1. ניהול קורסים

- **יצירה ועדכון**: **factory**: יצירה, עדכון ומחיקה של קורסים במערכת באמצעות מחלקה שיוצרת קורס לפי הסוג שלו **factory()**.
- **FlyWeight**: במידה והקורס שאנחנו רוצים ליצור כבר קיים, המערכת תחזיר את הקורס ולא תיצור אחד חדש.
- **Decorator**: לאחר יצירת קורס, המשתמש יכול להגדיר את הקורס כקורס מאסטר. (ציון סף מעבר של 80).

2. יצירת מערכת

- **יצירת מערכת**: **singleton**: המשתמש יוצר מערכת יחידה מולה כל העבודה מתנהלת.

3. רישום משתמשים

- **הוספת משתמשים**: הוספה של משתמשים וחיבורים למערכת.
- **רישום סטודנטים לקורס**: סטודנטים יכולים להירשם ולבטל קורסים.
- **התראות observer**: סטודנטים יכולים לבקש לקבל התראה כאשר בקורס מסוים שרצו, התפנה מקום להרשמה.
- **Façade**: קיימת מחלקה שמטרתה לפשט את הרישום של משתמש ולהסתיר מידע לא נחוץ לו.

4. אימות ובקרת גישה

- **אימות הרשאות**: בדיקת הרשאות לפני פעולות. (לדוגמא, סטודנטים אינם יכולים ליצור קורס).

שימוש בקוד

1. התקנה והרצה

- **הורדת קוד המקור**: להוריד את קוד המקור מהמאגר.
- **הרצת המערכת**: להריץ את הקוד בסביבת Java מותקנת.

2. פעולה

- **הפעלת ה main:** להפעיל את ה main-ולבדוק את פעולות המערכת בהתאם לצורך.

design pattern:

singleton: ישנה מערכת קורסים **אחת בלבד** ולא ניתן ליצור אחת נוספת. כל ניסיון ליצור אחת נוספת, יחזיר את המערכת שכבר קיימת.

observer: כאשר ניסיון של סטודנט נכשל עקב תפוסה מלאה בקורס, המערכת תשאל אותו אם הוא רוצה לקבל עדכון כאשר הקורס שוב זמין עבורו. במידה וכן, המערכת תוסיף אותו לרשימת הצופים ותבצע את ההתראה ברגע שהקורס יתפנה.

factory: כאשר מרצה או סטודנט רוצה ליצור קורס, אז מחלקת פקטורי תבצע את היצירה של הקורס ותיצור קורס מסוג ספציפי מתוך 3 סוגים שונים (סמינר, בחירה, חובה).

Flyweight: נמצא בתוך פקטורי. מחלקת פקטורי מחזיקה רשימת קורסים ובעת הקריאה לפונציה ליצור קורס, נבדוק קודם כל אם הקורס קיים כבר במערכת. אם קיים = נחזיר אותו. לא קיים = ניצור אחד חדש.

façade: בנינו מחלקה בשם RegistrationFacade. מחלקה זו אחראית על הוספה, הרשמה, התחברות, התנתקות של משתמשים וקורסים. מטרתה ליצור למשתמש ממשק פשוט ככל הניתן ולחשוף כמה שפחות מחלקות שאינן רלוונטיות אליו.

אמנם התבקשנו לבצע 5 תבניות עיצוב אך הוספתי לעצמי מתוך עניין ורצון ללמוד תבנית עיצוב נוספת

decorator: מחלקה בה אנחנו יכולים לקחת כל קורס קיים ולהפוך אותו לקורס 'מאסטר'. בקורס זה הציון מעבר יהיה 80 ולא 60 כמו שאר הקורסים הרגילים במערכת. נוכל לקבל את המידע באמצעות getDescription.