

Tematy projektów

Ogólne zasady

Zasady:

- Projekty oceniane na punkty w skali od 0 do 10.
- Tydzień spóźnienia z projektem -- maksymalnie 8 pkt, dwa tygodnie -- maksymalnie 6 pkt., potem 0 pkt.
- Treści zadań mogą ulec drobnym modyfikacjom mającym na celu jaśniejsze opisanie jak ma działać program do napisania.

Terminy

Etap 1: 11 grudnia godz 7:00

Etap 2: 24 grudnia godz 7:00

Etap 3: 10 stycznia godz 7:00

Wymagania

Temat 1: Rejestr pracowników

Napisz program oferujący następujące funkcjonalności:

- Program posiada klasę modelującą firmę informatyczną i jej pracowników.
- Firma pozwala na zatrudnianie i zwalnianie pracowników, wyświetlanie list zatrudnionych osób wraz z ich zarobkami.
- Pracownik ma metody do wykonywania pracy i picia kawy.

Etap 2:

Dodajemy klasę Stażysta, Administrator, Programista, Analityk, Księgowy. Pracownicy mogą być zatrudniani na różne rodzaje umów: o pracę, o dzieło lub na zlecenie, przez co inaczej będzie się liczyć ich wynagrodzenie. Program powinien umożliwiać przepracowanie miesiąca lub miesięcy, aby widać było wynagrodzenie poszczególnych osób. Osoby mogą iść na L4, wtedy ich pensja jest niższa i wynosi 0.7 wynagrodzenia z okresu przebywania na L4. Premia pracowników uzależniona jest od ilości wykonanej pracy, a pomniejszana jest o ilość wypitej kawy. Stażyści nie mają premii, ani nie chodzą na urlop. Programiści i analitycy mogą pracować zdalnie. Księgowi mogą pracować tylko na umowę o pracę, a Administratorzy mają więcej urlopu.

Temat 2: Fabryka samochodów

Napisz program oferujący następujące funkcjonalności:

- Program posiada klasę modelującą fabrykę pojazdów i produkowanych przez nią samochodów.
- Fabryka pozwala na konstrukcję nowych aut i ich sprzedawanie, wyświetlanie pojazdów wyprodukowanych (stanu fabryki) wraz z ich modelami.
- Samochód ma metody do tankowania i jazdy.

Etap 2:

Dodajemy klasy Motocykl, Rower (z opcjonalnym koszykiem), Pojazd i Pojazd Silnikowy,. Klasa Samochód i Motocykl mają informację o ilości paliwa, które pozwala przejechać daną liczbę kilometrów. paliwo można uzupełniać tankując samochód. Wszystkie Pojazdy posiadają daną ładowność, w przypadku roweru jest to zależne od posiadanego koszyka, a w klasie Samochodu czy Motocyklu jest to określone rozmiarem bagażnika w danej marce. Dodajemy klasę Komis, która posiada plac (np. vector) na Pojazdy Silnikowe. Komis może odkupić auto od właściciela lub mu je sprzedać. Cena skupu jest funkcją zależną od marki i przebiegu, a cena sprzedaży jest ceną zakupu z marżą zależną od komisu. Przy zakupie przez komis przebieg aut niektórych marek jest dzielony przez 2. Dodajemy klasę Lakiernik, która przyjmuje Pojazd do lakierowania i zmienia mu kolor.

Wymagania wspólne

Etap 1:

- Klasy posiadają przeładowany operator wypisywania na ekran.
- Program pozwala na wczytanie danych z pliku i zapis do niego, klasy posiadają przeładowane operatory operacji na strumieniach.
- Program oferuje menu prezentujące możliwości programu. Może być proste menu alfanumeryczne, CLI lub GUI (dopytać o biblioteki z jakich można korzystać).

Etap 2:

Program pozwala na uruchomienie z argumentem wiersza poleceń, którym jest ścieżka do pliku. W takim przypadku program ma wykonać polecenia (komendy) znajdujące się w tym pliku zamiast czytania ze standardowego wejścia. W przypadku braku argumentu wywołania programu, program zachowuje się jak dotychczas, czyli czyta komendy ze standardowego wejścia. W przesłanym archiwum proszę umieścić jeden lub więcej plików z komendami. Nazwa pliku powinna się zaczynać "stdin" i kończyć na ".txt".

Zaproponuj sposób dziedziczenia i oraz sposób rozmieszczenia pól składowych w tych klasach.

Etap 3:

- W programie znajdują się instrukcje rzucające wyjątki (throw) przynajmniej 2 typów oraz zapewniona jest ich obsługa. Obsługa sytuacji wyjątkowych znajduje się poza funkcją rzucającą wyjątek, tak aby instrukcja throw nie znajdowała się bezpośrednio w bloku try.
- W programie powinna się znaleźć przynajmniej jeden sensowny szablon klasy lub dwa szablony funkcji.
- Programy nie powinny wyświetlać żadnych ostrzeżeń podczas kompilacji.
- Pliki z komendami powinny prezentować nowe funkcjonalności, czyli pokazać wykorzystanie szablonów oraz rzucanie i łapanie wyjątków.
- Przy przekazywaniu argumentów do funkcji unikamy kopiowania obiektów.
- Można utworzyć więcej niż 1 fabrykę/firmę.