

Zadania do bloku GIT oraz Programowanie I - poziom podstawowy.

Tworzenie zadań tylko do GITa nie ma sensu więc połączyłem zagadnienia.

Napisz program pozwalający na rezerwację miejsc hotelowych. Projekt wykonuj etapowo. Wyniki swojej pracy zapisuj w repozytorium GITa.

Część I.

1. Utwórz nowe repozytorium HotelService a następnie wypchnij na nie pusty projekt (maven project z archetypu maven-archetype-quickstart). Pamiętaj o zignorowaniu niepotrzebnych plików.
2. Utwórz gałąź develop.
3. Wszystkie nowe funkcjonalności wprowadzaj na nowych gałęziach (najlepiej feature).
4. Dodaj klasę Room reprezentującą pojedynczy pokój (nr pokoju, ilu osobowy, czy w pokoju jest łazienka - true/false oraz czy pokój jest dostępny - true/false).
5. Dodaj klasę Hotel która będzie zawierała listę pokoi.
6. W konstruktorze klasy Hotel utwórz kilka obiektów klasy Room i dodaj je do listy (np. 10-15), dzięki temu podczas tworzenia instancji obiektu hotel, automatycznie generowana będzie lista pokoi.
7. Dodaj klasę UserService która będzie służyć do obsługi hotelu. Na początek dodaj funkcjonalności:
 1. Pobierz listę wszystkich pokoi.
 2. Pobierz listę wszystkich dostępnych pokoi.
 3. Rezerwuj pokój (podaj nr pokoju i jeśli jest dostępny to go zarezerwuj).
 4. Zwolnij pokój (podaj nr pokoju i jeśli jest zajęty to go zwolnij).
8. W klasie main utwórz proste menu do obsługi hotelu, przykłady implementacji (prosta pętla do...while + switch):
 1. <https://stackoverflow.com/a/13536215/5877109>
 2. <http://chronicles.blog.ryanrampersad.com/2011/03/text-based-menu-in-java/>
9. Do menu dodaj funkcję:
 1. Wyświetl listę pokoi wraz z ich statusem (wolny-zajęty) korzystając z HotelService.
 2. Wyświetl listę tylko dostępnych pokoi.
 3. Zarezerwuj pokój.
 4. Zwolnij pokój.
10. Zapisz i zrób commit na nowej gałęzi a następnie wykonaj merge swojej gałęzi z developem i usuń gałąź. Jeśli aplikacja działa bez zarzutów wykonaj także merge gałęzi develop do master (żadnej z nich jednak nie usuwaj).

Część II.

Dla kolejnych funkcjonalności twórz nowe gałęzie.

1. Dla pkt 1, 2 i 3 utwórz jedną nową gałąź. Dodaj klasę Guest która będzie reprezentowała hotelowego gościa (imię, nazwisko, data urodzenia).
2. Do klasy Room dodaj listę która będzie przechowywała gości zameldowanych w danym pokoju.
3. Metodę rezerwacji zmodyfikuj tak aby móc przekazać nr pokoju oraz gości którzy się w nim meldują. Zanim zameldujesz gości sprawdź czy przynajmniej 1 osoba jest pełnoletnia. Jeśli wszystko działa bez problemów, wykonaj merge swojej gałęzi do developa.
4. Dla pkt 4, 5 i 6 utwórz jedną nową gałąź. Do klasy Room dodaj parametr pozwalający zweryfikować (true/false) czy pokój jest posprzątany. Domyślnie wszystkie pokoje będą posprzątane, natomiast jeśli gość się wymelduje, pokój zmienia status na nieposprzątany. Nieposprzątanego pokoju nie można zarezerwować pomimo że jest wolny.
5. W klasie HotelService dodaj metodę pozwalającą posprzątać pokój (zmienić status pola posprzątanego na true). Metoda powinna przyjmować jako parametr nr pokoju.
6. W menu dodaj funkcję pozwalającą wyświetlić listę pokoi do posprzątania oraz funkcję posprzątania pokoju. Wykonaj merge do develop.
7. Dla pkt 7 i 8 utwórz jedną nową gałąź. W klasie Room dodaj pole data zameldowania oraz data wymeldowania.
8. W menu aplikacji dodaj funkcję pozwalającą wyświetlić listę zajętych pokoi wraz z datą opuszczenia go. Wykonaj merge do develop.
9. Jeśli wszystkie wyżej opisane funkcjonalności działają, wykonaj merge develop do master.