

Una compagnia di navigazione vuol realizzare un sistema di prenotazione on-line per l'imbarco su traghetti per la connessione con le isole. In particolare, il sistema dovrà permettere la prenotazione di biglietti per l'imbarco di persone e veicoli sia privati che commerciali. Una prenotazione può essere effettuata per una o più persone e relativi veicoli. Ogni prenotazione sarà caratterizzata da un orario d'imbarco, un orario di partenza/arrivo, la nave su cui imbarcarsi, la presenza o meno di un veicolo ed infine la tipologia di biglietti prenotati (privati o commerciali).

Definire un Class Diagram per modellare la prenotazione dei biglietti per l'imbarco di un gruppo di persone con 2 veicoli. Il Class Diagram, inteso come modello di dominio, si deve rifare all'Entity-Boundary-Control (EBC).

Una compagnia di navigazione vuol realizzare un sistema di prenotazione on-line per l'imbarco su traghetti per la connessione con le isole. In particolare, il sistema dovrà permettere la prenotazione di biglietti per l'imbarco di persone e veicoli sia privati che commerciali. Una prenotazione può essere effettuata per una o più persone e relativi veicoli. Ogni prenotazione sarà caratterizzata da un orario d'imbarco, un orario di partenza/arrivo, la nave su cui imbarcarsi, la presenza o meno di un veicolo ed infine la tipologia di biglietti prenotati (privati o commerciali).

In relazione all'esercizio precedente, fornire i Mock-up e un Sequence Diagram di analisi dello scenario relativo alla prenotazione dei biglietti per l'imbarco di un gruppo di persone con 2 veicoli. Il Sequence Diagram deve sfruttare le classi definite nel Class Diagram del punto precedente (che può essere parzialmente esteso in questo esercizio).

Si vuole realizzare un sistema informatico per semplificare la gestione delle finanze personali. Un utente registrato può specificare entrate mensili ricorrenti (e.g.: stipendio, pensione) e uscite mensili ricorrenti (e.g.: affitto, rata dell'automobile, abbonamento per i mezzi pubblici), indicando per ciascuna entrata/uscita ricorrente il corrispettivo importo in Euro e una breve descrizione. Inoltre, un utente può inserire nel sistema entrate e uscite occasionali (ovvero, non ricorrenti), indicando una data, una causale e un importo in Euro. Infine, un utente può visualizzare un bilancio provvisorio per il mese corrente. In questo caso, il sistema mostra all'utente il bilancio provvisorio calcolando la differenza tra le entrate del mese (sia ricorrenti che occasionali) e le uscite (sia ricorrenti che occasionali).

Definire un Class Diagram per modellare la visualizzazione del bilancio provvisorio del mese corrente. Il Class Diagram, inteso come modello di dominio, si deve rifare all'Entity-Boundary-Control (EBC).

Si vuole realizzare un sistema informatico per semplificare la gestione delle finanze personali. Un utente registrato può specificare entrate mensili ricorrenti (e.g.: stipendio, pensione) e uscite mensili ricorrenti (e.g.: affitto, rata dell'automobile, abbonamento per i mezzi pubblici), indicando per ciascuna entrata/uscita ricorrente il corrispettivo importo in Euro e una breve descrizione. Inoltre, un utente può inserire nel sistema entrate e uscite occasionali (ovvero, non ricorrenti), indicando una data, una causale e un importo in Euro. Infine, un utente può visualizzare un bilancio provvisorio per il mese corrente. In questo caso, il sistema mostra all'utente il bilancio provvisorio calcolando la differenza tra le entrate del mese (sia ricorrenti che occasionali) e le uscite (sia ricorrenti che occasionali).

In relazione all'esercizio precedente, fornire i Mock-up e un Sequence Diagram di analisi dello scenario relativo alla visualizzazione del bilancio provvisorio del mese corrente. Il Sequence Diagram deve sfruttare le classi definite nel Class Diagram del punto precedente (che può essere parzialmente esteso in questo esercizio).

```

public class Player {
    public static void main(String[] args) {
        Content c1 = new VideoContent();
        Content c2 = new AudioContent();
        c1.setNomeFile("Funny Cat Moments");
        c2.showDetails(); c1.showDetails();
    }
}

class Content {
    private String nomeFile;
    public void showDetails() {
        System.out.println("Nome File: "+nomeFile);
    }
    public void setNomeFile(String nome) {this.nomeFile=nome;}
}

class VideoContent extends Content {
    public void showDetails() {
        super.showDetails();
        System.out.println("Video File");
    }
}

class AudioContent extends Content {
    public void showDetails() {
        System.out.println(this.nomeFile);
        System.out.println("Audio File");
    }
}

```

Si consideri il codice Java sopra riportato.

- Il codice compila correttamente? Se no, indicare come modificare il codice affinché la compilazione abbia successo.
- Si consideri il codice eventualmente ottenuto dalle modifiche del punto precedente. Qual è l'output di un'invocazione del metodo main della classe Player?
- Si consideri il codice eventualmente ottenuto dalle modifiche dei punti precedenti. È presente un esempio di overloading? In caso di risposta positiva indicare in quale/quali parti di codice. In caso di risposta negativa, indicare – se possibile – come si potrebbe modificare ulteriormente il codice per introdurne un esempio.