

# Programmazione Orientata agli Oggetti

## Appello fittizio

### Parte 1: Quiz

Rispondere alle seguenti 8 domande cerchiando le risposte appropriate su questi fogli. **Scrivere nome, cognome e matricola su OGNI FOGLIO** che si consegna.

**Cosa caratterizza un tipo di dato astratto?** Scegli una o più alternative:

- a. il fatto che lo stato dell'oggetto sia modificabile
- b. il fatto che ci sia almeno un metodo che modifica lo stato
- c. le varie operazioni di accesso e modifica dello stato che esso supporta.

risposta: C

**Il pattern Builder serve per:** Scegli una o più alternative:

- a. costruire istanze di oggetti in maniera incrementale
- b. realizzare una gerarchia di tipi senza dover introdurre interfacce o classi astratte
- c. costruire istanze di oggetti mediante un meccanismo flessibile per specificare i valori di attributi
- d. rimpiazzare l'uso dei costruttori, che non vengono usati
- e. costruire degli oggetti base di un package.

risposta: A,C

**Il pattern Composite per funzionare richiede la ereditarietà multipla.**

- a. falso
- b. vero

risposta: A

**Per fare in modo che una interfaccia e le classi che l'implementano vengano compilate, è necessario definire gli invarianti relativi all'implementazione dell'interfaccia?**

- a. Falso
- b. Vero

risposta: A

**Cosa succede quando si compila o esegue il seguente codice? Scegli una o più alternative:**

```
public class Saluti {  
    public static void main(String[] args){  
        Uno o = new Uno();  
        o.saluta();}    }  
class Uno {  
    public final void saluta(){  
        System.out.println("Ciao");  
    }  
    public abstract void salutaPerBene(); }
```

- a. Errore di compilazione
- b. Scrive "ciao"
- c. Errore runtime
- d. Nessuna delle altre risposte

risposta: A

**Il seguente codice viene compilato correttamente?**

```
public class Contatore{  
    public int count(int i){  
        return 1;    }}  
class Contatore2 extends Contatore{  
    public int count(long i) throws Exception{  
        return -1;    }}
```

- a. Vero
- b. Falso

risposta: A

**Il principio della Segregazione delle Interfacce dice che:** Scegli una o più alternative:

- a. il contenuto di una interfaccia deve venir suddiviso tra varie classi astratte
- b. occorre avere una interfaccia per ogni classe concreta che si definisce
- c. occorre definire interfacce con pochi metodi
- d. una interfaccia utente va divisa in più pagine

risposta: C

**Un iteratore è un tipo di dato astratto?**

- a. vero

b. falso

risposta: A

## Parte 2: Sviluppo di un programma Java

Considerare la situazione descritta sotto e implementare tipi e metodi in Java per supportare le funzionalità richieste. È importante che si specifichino adeguatamente tipi di dati astratti e contratti e che si ottemperi al principio di singola responsabilità. L'adozione di "clean coding" e di pattern di design ove appropriato è pure importante. Escludere ogni problematica relativa alla interfaccia utente e alla persistenza.

### RentACar

RentACar è un sistema di gestione per un'azienda di noleggio veicoli che opera con una flotta diversificata composta da automobili, furgoni e bus. L'azienda dispone di più stazioni di raccolta dove i clienti possono iniziare e terminare i loro noleggi. Il sistema si occupa di gestire la disponibilità dei veicoli, monitorare i noleggi in corso e calcolare i costi in base a molteplici parametri.

Il sistema deve gestire in memoria (senza persistenza o interfaccia utente) le seguenti informazioni e operazioni: gestione della flotta di veicoli, monitoraggio dei noleggi attivi, calcolo dei costi e analisi delle prestazioni del servizio.

Progettare e realizzare una serie di classi Java che supporta, tra le altre, le seguenti operazioni:

1. Verificare la disponibilità di un veicolo per un noleggio, specificando:

- Tipo di veicolo richiesto (auto, furgone, bus)
- Modello desiderato
- Stazione di ritiro
- Data e ora di inizio
- Durata prevista
- Stazione di riconsegna prevista

2. Calcolare il costo del noleggio considerando:

- Tariffa base per tipo e modello del veicolo
- Durata del noleggio
- Chilometraggio stimato
- Eventuali differenze tra stazione di ritiro e riconsegna
- Possibili sovrapprezzi per modifiche durante il noleggio

3. Avviare un noleggio registrando:

- Dati del cliente
- Veicolo selezionato
- Stazione di ritiro e di consegna
- Data e ora di inizio, data e ora stimata di fine

- Condizioni concordate

4. Disporre di una dashboard che presenta informazioni sui noleggi in corso

Produrre anche il codice del Main che illustri esempi di chiamata dei metodi che implementano queste operazioni.