

Esercizio Gerarchia Animali

Obiettivo:

Creare una gerarchia di classi e interfacce che rappresentano diversi tipi di animali, applicando i concetti di polimorfismo, ereditarietà, e implementazione libera di classi astratte o interfacce.

Descrizione del Compito:

Realizzare un sistema in Java che gestisca diversi animali con comportamenti comuni e specifici. Ogni studente può decidere liberamente come utilizzare classi astratte o interfacce per organizzare il codice, seguendo le indicazioni seguenti:

1. Definire comportamenti comuni a tutti gli animali, come:
 - `emettiVerso()` per produrre il suono caratteristico.
 - `muoviti()` per descrivere il movimento dell'animale.
 - `mostrainfo()` per visualizzare informazioni generali sull'animale.
2. Creare almeno tre tipi specifici di animali (esempio: `Cane`, `Uccello`, `Pesce`), ognuno con comportamenti specifici diversi implementando il metodo `emettiVerso()` e `muoviti()` in modo appropriato.
3. Aggiungere almeno un comportamento specifico unico per ciascun animale (ad esempio, il cane può avere il metodo `riportaOggetto()`, l'uccello `vola()`, il pesce `nuota()` ecc.).
4. Realizzare una classe di gestione `Zoo` o simile che:
 - Gestisca un array di animali in maniera polimorfica.
 - Permetta l'aggiunta di nuovi animali.
 - Visualizzi informazioni dettagliate di tutti gli animali presenti.
5. Creare una classe `TestAnimali` per:
 - Istanziare almeno un oggetto di ciascun tipo di animale.
 - Gestire gli oggetti attraverso la classe `Zoo`.
 - Richiamare i metodi comuni (`emettiVerso`, `muoviti`, `mostrainfo`) e quelli specifici definiti per ciascun animale.

Istruzioni per lo Svolgimento:

1. Scrivere il codice Java rispettando i requisiti sopra elencati.
2. Eseguire e testare accuratamente il programma.
3. Zippare tutte le classi create (almeno 5 inclusa la classe di gestione e il test) in un file `.zip`.
4. Caricare il file zip nella cartella `EserciziStudenti`.
5. Naming Convention: `EsercizioAnimali - Gruppo.zip`.