

Z00: una collezione di funzioni per la gestione di animali di uno zoo

Alessandro Bocci, Susanna Pelagatti

Primo assegnamento Python in itinere 622AA
AA 2024-25

Indice

1	Introduzione	1
1.1	Materiale in linea	1
1.2	Struttura degli assegnamenti, bonus, tempi di consegna e prova orale	1
1.3	Consegna degli assegnamenti	2
1.4	Valutazione dell'assegnamento	2
2	L'assegnamento Z00	2
2.1	Istruzioni	3

1 Introduzione

Il primo modulo (9 crediti) del corso di Programmazione e Analisi Dati (622AA) per la parte Python prevede lo svolgimento di tre assegnamenti in itinere (uno Scratch e due Python) che esonerano dallo svolgimento del progetto finale. Questo documento descrive il primo assegnamento Python che prevede la realizzazione di un insieme di funzioni Python.

Il software viene sviluppato e documentato utilizzando gli strumenti, le tecniche e le convenzioni presentati durante il corso.

1.1 Materiale in linea

Tutto il materiale relativo al corso può essere reperito dal Moodle ufficiale del corso. Eventuali chiarimenti possono essere richiesti alle docenti per posta elettronica.

1.2 Struttura degli assegnamenti, bonus, tempi di consegna e prova orale

I tre assegnamenti possono essere svolti individualmente o in gruppi di 2 studenti e possono essere consegnati entro il 31 Gennaio 2025. Gli studenti che consegnano una realizzazione sufficiente entro la data stabilita dai docenti per

ciascun assegnamento avranno un incremento di 2 punti (di Bonus) nella valutazione complessiva di ciascun assegnamento. La valutazione complessiva viene data dalla media delle valutazioni dei tre assegnamenti e costituisce la base per la prova orale. La prova orale sarà composta di due parti. La prima parte sarà una discussione sugli assegnamenti consegnati e tenderà a stabilire se lo studente è realmente l'autore di quanto consegnato (in caso di dubbi la valutazione verrà opportunamente aggiustata) in particolare verrà chiesto di leggere e modificare il codice e di spiegare quanto usato (in particolare costrutti e moduli non facenti parte del programma del corso). La seconda parte verterà su tutto il programma del corso. In particolare, l'orale comprenderà:

- una discussione delle scelte implementative
- l'impostazione e la scrittura di semplici programmi Python (sequenziali e GUI) di difficoltà medio bassa rispetto a quelli visti nelle esercitazioni in classe
- domande su tutto il programma presentato durante il corso.

Il voto finale comprenderà la valutazione effettiva degli assegnamenti (emendata in caso di dubbi) e la prova orale.

1.3 Consegna degli assegnamenti

La consegna degli assegnamenti avviene *esclusivamente* attraverso il Moodle del corso, seguendo le istruzioni nei file `README` di ciascun assegnamento.

1.4 Valutazione dell'assegnamento

All'assegnamento viene assegnata una fascia di valutazione da 0 a 30 che tiene conto dei seguenti fattori:

- motivazioni, originalità ed economicità delle scelte implementative
- strutturazione del codice (fattorizzazione del codice in funzioni, uso di strutture dati adeguate etc)
- efficienza e robustezza (numero di operazioni eseguite, fallimenti in caso di input inadeguati etc)
- aderenza alle specifiche
- qualità del codice Python e dei commenti

Tutti gli assegnamenti verranno confrontati automaticamente per verificare situazioni di plagio. Nel caso di elaborati uguali verranno presi provvedimenti per tutti i gruppi coinvolti.

2 L'assegnamento ZOO

L'assegnamento prevede la realizzazione di un insieme di funzioni per la gestione semplificata di animali vertebrati presenti in uno zoo. Ogni animale è rappresentato da una tupla:

(nome, classe, specie, numero_zampe, sangue_caldo, ambiente, zona, verso)

I campi `nome`, `classe`, `specie` e `verso` sono tutti di tipo stringa. Il campo `numero_zampe` è di tipo intero. Il campo `sangue_caldo` è di tipo booleano. Il campo `ambiente` è di tipo lista.

Il campo `classe` deve essere uno fra “Mammifero”, “Uccello”, “Rettile”, “Anfibio” o “Pesce”. Il campo `ambiente` deve comprendere almeno uno fra “Acqua”, “Terra” e “Aria”. Il campo `zona` è composto da una lettera e un numero intero di una cifra atti a rappresentare la zona dello zoo in cui l’animale vive.

Una tupla di esempio per un animale è:

(“Alex”, “Mammifero”, “Leone”, 4, True, [“Terra”], “A5”, “Ruggito”)

L’insieme degli animali è rappresentato come un dizionario con chiave il nome il quale deve essere unico per ogni animale.

Devono essere realizzate (almeno) le funzioni specificate nel file `zoo.py` (leggere il file per le specifiche dettagliate).

È possibile definire funzioni ausiliarie per la soluzione ed estendere le funzionalità del catalogo definendo nuove funzioni.

Per ogni funzione, verificare che i tipi dei parametri passati durante la chiamata siano quelli attesi e segnalare le situazioni erranee con opportuni messaggi di errore stampati a schermo.

2.1 Istruzioni

Unitamente a questo file vengono forniti due file Python contenenti l’intestazione delle funzioni da realizzare (`zoo.py`) e i test che devono essere superati per consegnare il codice (`main.py` e `testMy.py`) e un file di istruzioni (`README`).

Consigliamo di leggere attentamente le istruzioni e analizzare il codice fornito per capire come strutturare le funzioni prima di iniziare a progettare e implementare. Ricordiamo l’importanza di analizzare i vari casi prima di iniziare a scrivere codice e di effettuare test incrementali sul codice durante lo sviluppo.

Possono essere realizzate funzionalità in più rispetto a quelle richieste (ad esempio altre funzioni).

Le parti opzionali devono essere corredate da test appropriati e documentate da commenti chiari o (in caso sia necessario) da un breve documento descrittivo che può essere consegnato insieme al codice e che spiega le motivazioni e la struttura di quanto realizzato.