

Esercitazione Python del 11 ottobre 2016

Obiettivo dell'esercitazione è di prendere confidenza con Python e con il suo ambiente IDLE. Per risolvere gli esercizi descritti di seguito dovete usare l'editor di IDLE che si apre cliccando sull'icona di Python sul desktop e poi selezionando File → New File. Scrivete in questo file la vostra soluzione e poi salvatela sul desktop. Usate file diversi per i vari esercizi, ad esempio chiamando i file "esercizio1.py", "esercizio2.py" e così via. Ricordatevi di copiare i file sulla vostra pennetta USB prima di andare via.

Funzioni Utili

Per risolvere gli esercizi possono essere utili queste funzioni Python:

- **len(stringa)** restituisce la lunghezza della stringa

I metodi delle stringhe utili per risolvere gli esercizi sono applicati, ad esempio, a una stringa s:

- **s.replace(str1, str2, 2)** crea una nuova stringa partendo da s e sostituendo le prime 2 occorrenze di str1 con str2. Notate che str1 e str2 possono essere anche stringhe di lunghezza qualsiasi e che se si vuole sostituire tutte le occorrenze allora il numero si può omettere, ad esempio s.replace("ab", "bac")
- **s.find(str)** calcola la prima posizione di s in cui compare la stringa str. Se la stringa str non compare in s restituisce -1
- **s.rfind(str)** calcola la prima posizione di s in cui compare la stringa str partendo dalla fine della stringa s. Se la stringa str non compare in s restituisce -1
- **s.count(str)** calcola quante volte str compare in s
- **s[i]** restituisce il carattere di indice i (posizione i+1)
- **s[-1]** restituisce l'ultimo carattere
- **s[i:k]** restituisce la sottostringa tra gli indici i e k-1
- **s[i:]** restituisce la sottostringa tra l'indice i e la fine
- **s[:k]** restituisce la sottostringa tra l'inizio e l'indice k-1
- **chr(intero)** restituisce il carattere con il codice intero
- **ord(carattere)** restituisce il codice del carattere

Esercizi

- 1) Scrivere un programma che legge in input una stringa e un intero e stampa il carattere la cui *posizione* corrisponde all'intero. Ad esempio se la stringa è e 'Paperopoli' e l'intero è 4 il programma deve stampare 'e'. Si noti che l'*indice* di 'e' vale 3;
- 2) Scrivere un programma che legge due stringhe e stampa la somma delle loro lunghezze. Ad esempio se le stringhe sono 'casa' e 'bella' il programma deve stampare 9;
- 3) Scrivere un programma che prende in ingresso una stringa s lunga almeno tre caratteri e stampa la stringa senza il primo e l'ultimo carattere. Ad esempio, se la stringa inserita è "Roma" il programma deve stampare "om". Se la stringa è più corta di tre caratteri il programma deve stampare un messaggio di errore;
- 4) Scrivere la funzione **successivo(s)** che prende in ingresso una stringa s e restituisce la stringa in cui il primo carattere è sostituito dal carattere successivo nell'alfabeto. Ad esempio, se la stringa inserita è "Roma" il programma deve restituire "Soma";
- 5) Scrivere una funzione **uguali(s)** che prende in ingresso una stringa s e verifica se il primo e l'ultimo carattere sono uguali. In caso positivo restituisce True altrimenti False;
- 6) Scrivere un programma che legge in input 3 numeri interi , **n1**, **n2**, **n3**, e li stampa in ordine decrescente. Assumete che i numeri siano tutti diversi tra di loro.
- 7) Scrivere un programma che legge in ingresso una stringa s e sostituisce tutte le occorrenze (tranne la prima) delle prime 2 lettere con "__". Ad esempio, se la stringa è "barabba", il programma deve stampare "barab__"
- 8) Scrivere un programma che prende in ingresso 3 interi **a**, **b**, **c**, corrispondenti ai lati di un triangolo e controlla la correttezza dei dati e stampa il tipo del triangolo (scaleno, isoscele, equilatero)
- 9) Scrivere un programma che prende in ingresso una stringa s corrispondente al nome di un file su disco e stampa la stringa senza l'eventuale suffisso. Ad esempio se s vale "primo.programma.py" il programma deve stampare "primo.programma". Testare il programma anche con un file senza suffisso.