



سوال ۱ برنامه‌ای بنویسید که ۱۰ نام از ورودی دریافت کند و اسامی که با حروف a یا A شروع می‌شوند را در یک لیست قرار دهد و چاپ کند.

```
# Function to get and print names that start with 'a' or 'A'
def get_and_print_a_names():
    names = []
    a_names = []

    # Get 10 names from input
    for i in range(10):
        name = input("Enter a name: ")
        names.append(name)

    # Place names that start with 'a' or 'A' in the a_names list
    for name in names:
        if name.lower().startswith("a"):
            a_names.append(name)

    # Print names
    print("Names that start with 'a' or 'A':")
    for a_name in a_names:
        print(a_name)

# Execute the function
get_and_print_a_names()
```





سوال ۲ برنامه‌ای بنویسید که یک لیست از اعداد را از ورودی دریافت کند. سپس یک عدد جدید را از کاربر بگیرد و تعداد تکرار آن عدد و محل‌های (ایندکس‌های) وجود آن عدد را در خروجی چاپ کند.

```
# Get list of numbers from input
k = input("Enter numbers separated by space: ").split()

# Convert the strings in the list to integers
for i in range(len(k)):
    k[i] = int(k[i])

# Get the new number from the user
number = int(input("Enter a number: "))

# Count the occurrences of the number in the list
count = k.count(number)

# Find the indices of the number in the list
indices = []
for i in range(len(k)):
    if k[i] == number:
        indices.append(i)

# Print the results
print("Count:", count)
print("Indices:", indices)
```





سوال ۳ : برنامه‌ای بنویسید که تعداد دانشجویان را از ورودی دریافت کند. سپس برای هر دانشجو شماره دانشجویی و نام را دریافت کند. در نهایت مشخصات تمام دانشجویان را چاپ کند.

```
class Student:
    def __init__(self, student_id, name):
        self.student_id = student_id
        self.name = name

    def is_valid_id(self):
        return len(self.student_id) == 8

    def __str__(self):
        return f"ID: {self.student_id}, Name: {self.name}"

# Get the number of students from input
num_students = int(input("Enter number of students: "))

students = []

# Get the details of each student
for _ in range(num_students):
    student_id = input("Enter student ID: ")
    name = input("Enter student name: ")
    student = Student(student_id, name)
    students.append(student)

# Print the details of the students
for student in students:
    if student.is_valid_id():
        print(student)
    else:
        print(f"Invalid ID for {student.name}")
```





```
def test(m):  
    if len(m) == 1:  
        return m[0]  
    else:  
        print(list)  
        return m[0] + test(m[1:])  
  
k = [10, 20, 9, 5]  
print(test(k))
```

این برنامه پایتون یک تابع به نام test تعریف می کند که یک لیست m را به عنوان ورودی می گیرد و مجموع اعضای لیست را برمی گرداند. اگر طول لیست ورودی ۱ باشد، تابع مقدار اولین عنصر لیست را برمی گرداند. در غیر این صورت، مقدار اولین عنصر لیست به جمع تابع فراخوانی می شود با استفاده از باقی عناصر لیست (بریده شده) به عنوان ورودی.

خروجی `print(k(test))`:

k به عنوان ورودی به print داده می شود.

اما print انتظار یک ورودی نوع متنی دارد و تابع test به عنوان ورودی از نوع list است. بنابراین، اینجا یک خطا از نوع TypeError رخ می دهد.





سوال ۵: خروجی تمامی پرینت های برنامه زیر را به ترتیب بنویسید و در نهایت کار تابع را را توضیح دهید :

Class variable : این متغیر عنوان یک متغیر کلاس تعریف شده است. این متغیر **بین تمام نمونه های کلاس مشترک است**.

Instance variables : این متغیرها عنوان متغیرهای نمونه در متد `__init__` تعریف شده اند و **منحصربه فرد هستند**.

نمونه کد در صفحه بعد



```
class Dog:
    # Class variable
    species = "Canis familiaris"

    def __init__(self, name, age):
        # Instance variables
        self.name = name
        self.age = age

    def describe(self):
        return f"{self.name} is {self.age} years old."

    def speak(self, sound):
        return f"{self.name} says {sound}."

# Creating instances of the Dog class
dog1 = Dog("Buddy", 5)
dog2 = Dog("Milo", 3)

# Accessing class variable
print(Dog.species) # Output: Canis familiaris
print(dog1.species) # Output: Canis familiaris
print(dog2.species) # Output: Canis familiaris

# Accessing instance variables
print(dog1.describe()) # Output: Buddy is 5 years old.
print(dog2.describe()) # Output: Milo is 3 years old.

# Modifying instance variables
dog1.age = 6
print(dog1.describe()) # Output: Buddy is 6 years old.

# Modifying class variable
Dog.species = "Canis lupus"
print(Dog.species) # Output: Canis lupus
print(dog1.species) # Output: Canis lupus
print(dog2.species) # Output: Canis lupus

# Changing class variable through instance
dog1.species = "New Species"
print(dog1.species) # Output: New Species (changes only for this instance)
print(dog2.species) # Output: Canis lupus (unchanged for other instances)
print(Dog.species) # Output: Canis lupus (unchanged for the class itself)
```





سوال ۶: برنامه‌ای بنویسید که دو لیست را از ورودی بگیرد و بررسی کند که آیا آن‌ها حداقل یک عضو مشترک دارند یا خیر.

```
# Get two lists from input
list1 = input("Enter first list of numbers separated by space: ").split()
list2 = input("Enter second list of numbers separated by space: ").split()

# Convert the strings in the lists to integers
for i in range(len(list1)):
    list1[i] = int(list1[i])

for i in range(len(list2)):
    list2[i] = int(list2[i])

# Check for common elements
common_found = False

for num1 in list1:
    for num2 in list2:
        if num1 == num2:
            common_found = True
            break
    if common_found:
        break

# Print the result
if common_found:
    print("Yes, they have at least one common element.")
else:
    print("No, they do not have any common elements.")
```





سوال ۷: تابعی بنویسید که عددی را به عنوان ورودی دریافت کند و بررسی کند که آیا آن عدد شامل رقم صفر است یا خیر.

```
def has_zero(n):  
    # Convert the number to string and check for '0'  
    if "0" in str(n):  
        return "The number contains zero."  
    else:  
        return "The number does not contain zero."  
  
# Get the number from the user  
number = int(input("Enter a number: "))  
print(has_zero(number))
```

