```
10 =""" Polymorphism and Inheritance: Write a python program to find
the whether the given
input is palindrome or not (for both string and integer) using the
concept of
polymorphism and inheritance."""
print(Q10)

# Program using Polymorphism and Inheritance to check Palindrome for
String and Integer
# Base class
class PalindromeChecker:
  def is_palindrome(self, value):
    pass # To be overridden by derived classes
# Derived class for String Palindrome
class StringPalindrome(PalindromeChecker):
  def is_palindrome(self, value):
    value = value.lower()
    return value == value[::-1]

# Derived class for Number Palindrome
class NumberPalindrome(PalindromeChecker):
  def is_palindrome(self, value):
    try:
        value = int(value)  # Ensure value is an integer
        value = str(value)
        return value == value[::-1]
    except ValueError:
        return False # Return False for invalid integer input

# Create objects for both classes
str_checker = StringPalindrome()
num_checker = NumberPalindrome()

# Input from user
string_input = input("Enter a string: ")

while True:
    num_input_str = input("Enter an integer: ")
    try:
        num_input = int(num_input_str)
        break  # Exit loop if valid integer is entered
    except ValueError:
        print("Invalid input. Please enter an integer.")


# Check for palindrome
if str_checker.is_palindrome(string_input):
  print(f"'{string_input}' is a palindrome string.")
else:
```

```
  print(f"'{string_input}' is not a palindrome string.")

if num_checker.is_palindrome(num_input):
  print(f"{num_input} is a palindrome number.")
else:
  print(f"{num_input} is not a palindrome number.")
```

Polymorphism and Inheritance: Write a python program to find the whether the given
input is palindrome or not (for both string and integer) using the concept of
polymorphism and inheritance.
Enter a string: program
Enter an integer: 55755
'program' is not a palindrome string.
55755 is a palindrome number.
