

Dealer Name: Ford

Dealer Info: Car Manufacturer

Dealer Number: FRD6787O

Question: palindrome code

Answer: ```python

```
def is_palindrome(text):
```

```
    """
```

Checks if a given string is a palindrome.

Args:

text: The string to check.

Returns:

True if the string is a palindrome, False otherwise.

```
    """
```

```
# Remove spaces and convert to lowercase
```

```
text = text.replace(" ", "").lower()
```

```
# Check if the string is the same forwards and backwards
```

```
return text == text[::-1]
```

```
# Example usage
```

```
text1 = "racecar"
```

```
text2 = "A man, a plan, a canal: Panama"
```

```
text3 = "hello world"
```

```
print(f'{text1} is a palindrome: {is_palindrome(text1)}')
```

```
print(f'{text2} is a palindrome: {is_palindrome(text2)}')
```

```
print(f'{text3} is a palindrome: {is_palindrome(text3)}')
```

```
```
```

## Explanation:

### 1. Function Definition:

- `def is_palindrome(text):`: defines a function named `is_palindrome` that takes a string `text` as input.

### 2. Preprocessing:

- `text = text.replace(" ", "").lower()`: This line removes any spaces from the string and converts it to lowercase. This ensures that the comparison is case-insensitive and ignores spaces.

### 3. Palindrome Check:

- `return text == text[::-1]`: This is the core logic. It compares the original string (`text`) with its reversed version (`text[::-1]`). If they are the same, the string is a palindrome, and the function returns `True`. Otherwise, it returns `False`.

### 4. Example Usage:

- Three example strings are defined: `text1`, `text2`, and `text3`.  
- The `is_palindrome` function is called for each string, and the result is printed along with the original string.

## How it works:

- The `[::-1]` slicing technique in Python reverses the string.
- The code compares the original string with the reversed string.
- If they match, the string is a palindrome, and `True` is returned.
- If they don't match, the string is not a palindrome, and `False` is returned.

This code provides a straightforward and efficient way to check for palindromes in strings.