

## Code

```
public partial class Page1 : Page
{
    public Page1()
    {
        if (DateTime.Now.Ticks % 10 == 0)
            _value = "Test";
        Debug.WriteLine($"The value is {_value}");
    }
    ~Page1()
    {
        if (_value.Equals("Test"))
            _value = string.Empty;
        Debug.WriteLine($"The value is {_value}");
    }
}
```

## Possible Outputs

Case	_value	DateTime.Now.Ticks	% 10 == 0	Output of the Constructor	Output of the Destructor
01	null	638321161892240791	false	The value is	Exception: _value is null
02	null	638321161892240670	true	The value is Test	The value is
03	string.Empty	638321161892240835	false	The value is	The value is
04	string.Empty	839328161852240770	true	The value is Test	The value is
05	“anything”	437321961896244886	false	The value is anything	The value is anything
06	“anything”	134328161859240000	true	The value is Test	The value is

---

## Explanation

- **Case 01**

If the `_value` is null, and `DateTime.Now.Ticks` is not divisible by `10` i.e. `DateTime.Now.Ticks % 10 == 0` is false, then accessing the `_value.Equals` in the destructor causes the Exception.

- **Case 02**

If the `_value` is null, and `DateTime.Now.Ticks` is indeed divisible by `10` i.e. `DateTime.Now.Ticks % 10 == 0` is true, then `string.Empty` is assigned to uninitialized `_value`, and accessing the `_value.Equals` in the destructor doesn't cause the Exception.

- **Case 03 and 05**

If the `_value` is initialized with some value, and `DateTime.Now.Ticks` is not divisible by `10` i.e. `DateTime.Now.Ticks % 10 == 0` is false, then the output from `public Page1()` and `~Page1()` is as listed in the table.

- **Case 04 and 06**

If the `_value` is initialized with some value, and `DateTime.Now.Ticks` is indeed divisible by `10` i.e. `DateTime.Now.Ticks % 10 == 0` is true, then the output from `public Page1()` and `~Page1()` is as listed in the table.