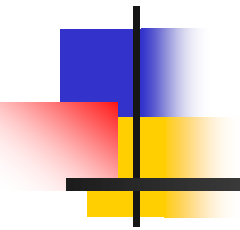
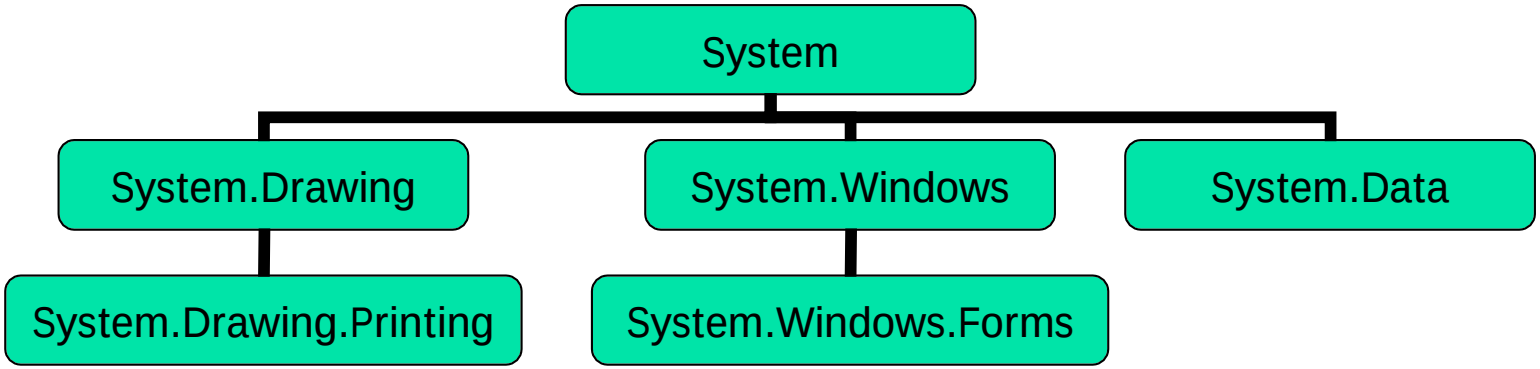
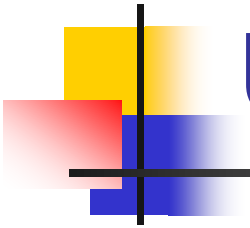


Selecting Alternatives in Visual C#.NET

by Derek Peacock

Namespaces

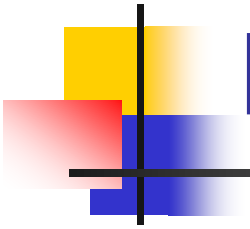

System
System.Drawing
System.Windows
System.Data
System.Drawing.Printing
System.Windows.Forms



Using

- **using** System.Convert;
- **using** System.Math;
- **using** System.DateTime;

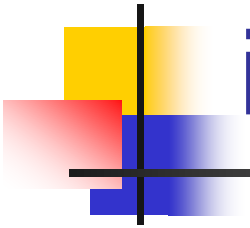
- using System;



Declaring Dates

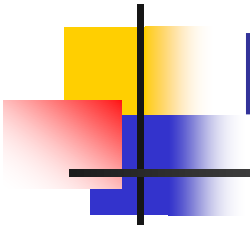
```
private DateTime today = DateTime.Today
private DateTime nextDate;

nextDate = today.AddDays(1)
nextDate = today.AddMonths(-2)
nextDate = today.AddYears(1)
```



if Statement

```
if (condition == true)
   statement1;
else
    statement2;
```

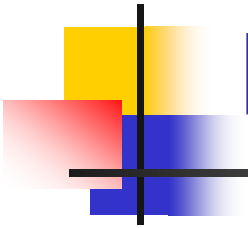


Example if

```
if(salary > 42000)
{
    taxRate = 40.0;
}
else
   taxRate = 25.0;
```

Relational Operators

Operator	Meaning
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
==	Equal to
!=	Not equal to



Boolean Expressions

(colour != Color.Red)

(animal == cat)

(cost <= 12.50)

(name < “Gita”)

(mark >= 85)

AND Logical Operator

Expression A	Expression B	A AND B
False	False	False
False	True	False
True	False	False
True	True	True

OR Logical Operator

Expression A	Expression B	A OR B
False	False	False
False	True	True
True	False	True
True	True	True

XOR Logical Operator

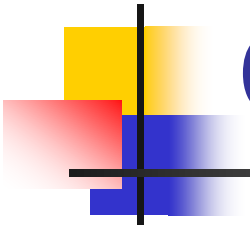
Expression A	Expression B	A XOR B
False	False	False
False	True	True
True	False	True
True	True	False

NOT Logical Operator

Expression A	NOT A
False	True
True	False

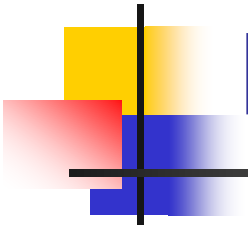
Precedence Order

1	Brackets, NOT	(), !
2	Arithmetic Operators	(++, --), (*/%), (+-)
3	Relational Operators	(<, >, <=, >=), (==, !=)
4	Logical Operators	(&&), (||)



Good Practice

- Keep Boolean and Arithmetic expressions as simple as possible
- Split the expressions to make it easier to understand
- ALWAYS USE ( )



Boolean Expression Quiz

1. (4 >= 5) && (8 == (3 + 5))
2. (4 >= 5) || (8 == (3 + 5))
3. (-2 < 0) && (18 >= 10)
4. ! (18 == (10 + 8))
5. ! (4 < 5) && (8 == (3 + 5))
6. ! ((4 < 5) && (8 == (3 + 5)))

Char Data Type

```
public string getCommand()
{
        string command = "None";
        var reply = Console.ReadKey();

        ‘Continued on next slide
```

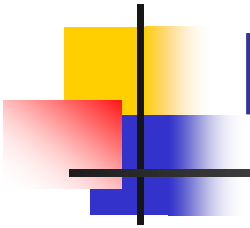
else if Statement

```
if (reply.KeyChar == 'L')
    command = "Load";
else if (reply.KeyChar == 'S')
    command = "Save";
else if (reply.KeyChar == 'Q')
    command = "Quit";


return command;
```

switch case Statement

```
switch (reply.KeyChar)
{
    case 'L':  case 'l':
          command = "Load";
          break; // or return
    case 'S':  case 's':
          return "Save";
    default:
          command = "Error";
          break;
}
```



References

- “Microsoft Visual C# Step by Step” by John Sharp (2014) Microsoft Press £19 (£14 Kindle)
- Chapter 4