



C# Cheat Sheet – Podstawy



Zmienne i typy danych



Zmienne

```
int liczba = 42;  
double temperatura = 36.6;  
string nazwa = "Czujnik";  
bool aktywny = true;
```



Stałe (const)

```
const double MaxTemperatura = 100.0;
```



Pętle



for

```
for (int i = 0; i < 10; i++)  
{  
    Console.WriteLine(i);  
}
```



foreach

```
foreach (var item in lista)  
{  
    Console.WriteLine(item);  
}
```



while

```
int i = 0;  
while (i < 5)  
{  
    Console.WriteLine(i);  
}
```

```
    i++;  
}
```

do-while

```
int licznik = 0;  
do  
{  
    Console.WriteLine($"Licznik: {licznik}");  
    licznik++;  
}  
while (licznik < 5);
```

Instrukcje warunkowe

```
if (temperatura > 50)  
{  
    Console.WriteLine("Alarm!");  
}  
else if (temperatura > 30)  
{  
    Console.WriteLine("Uwaga!");  
}  
else  
{  
    Console.WriteLine("OK");  
}  
  
switch (status)  
{  
    case "OK":  
        break;  
    case "Błąd":  
        break;  
}
```

Klasy i obiekty

```
public class Sensor  
{  
    public string Name { get; set; }  
    public double Value { get; set; }  
}
```

```

    public void Read()
    {
        Console.WriteLine("Odczyt danych...");
    }
}

Sensor s = new Sensor();
s.Read();

```



Dziedziczenie

```

public class TemperatureSensor : Sensor
{
    public string Unit { get; set; }

    public void Calibrate()
    {
        Console.WriteLine("Kalibracja czujnika temperatury");
    }
}

TemperatureSensor ts = new TemperatureSensor();
ts.Name = "TS-01";
ts.Value = 21.5;
ts.Calibrate();

```



Kolekcje



Lista (List<T>)

```

var lista = new List<string>();
lista.Add("A");
lista.Remove("A");

```



Słownik (Dictionary<TKey, TValue>)

```

var dict = new Dictionary<string, int>();
dict["czujnik1"] = 42;
int wartosc = dict["czujnik1"];

```



Kolejka (Queue<T>)

```
var kolejka = new Queue<string>();  
kolejka.Enqueue("Zadanie1");  
var pierwszy = kolejka.Dequeue(); // "Zadanie1"
```

Stos (Stack<T>)

```
var stos = new Stack<string>();  
stos.Push("Krok1");  
var ostatni = stos.Pop(); // "Krok1"
```

LINQ – filtrowanie, sortowanie, grupowanie

```
var aktywne = czujniki.Where(x => x.Aktywny);  
var posortowane = czujniki.OrderBy(x => x.Temperatura);  
var tylkoNazwy = czujniki.Select(x => x.Name);  
var grupy = czujniki.GroupBy(x => x.Typ);
```

Obsługa błędów

Blok try-catch finally

```
try  
{  
    // ryzykowna operacja  
}  
catch (Exception ex)  
{  
    Console.WriteLine(ex.Message);  
}  
finally  
{  
    // wykonywane zawsze  
}
```

Własny wyjątek

```
public class OverheatException : Exception  
{  
    public OverheatException(string message) : base(message) { }  
}
```

```
// użycie:  
double temperatura = 105;  
  
if (temperatura > 100)  
{  
    throw new OverheatException("Temperatura przekroczyła 100°C!");  
}
```



Praca z plikami

```
File.WriteAllText("log.txt", "Start systemu");  
string zawartosc = File.ReadAllText("log.txt");
```



Serializacja JSON

```
var json = JsonSerializer.Serialize(sensor);  
var obj = JsonSerializer.Deserialize<Sensor>(json);
```
