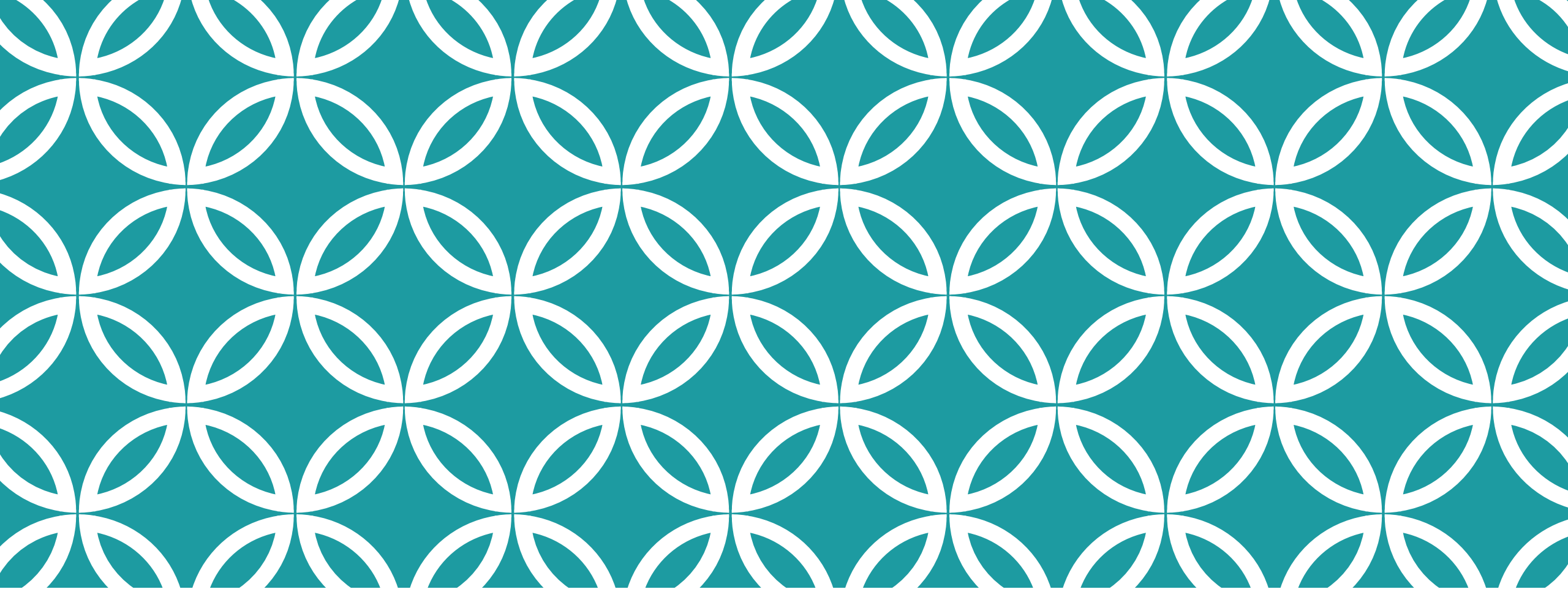


UML

**OBJEKT ORIENTERET PROGRAMMERING
GRUNDLÆGGENDE BEGREBER
MANGE SPROG**

Grundlæggende
programmering
Lektion 2



UML

Unified Model Language

UML

Repetition

Fortæl hvad et usecase diagram er og tegn det på tavlen

Fortæl hvad et klasse diagram er og tegn det på tavlen

UML

STRUKTUR DIAGRAMMER

Klasse diagrammer

En klasse repræsenterer en gruppe af ting, der har fælles tilstand og adfærd

En klasse er repræsenteret af en rektangulær kasse opdelt i rum

- Første rum indeholder navnet på klassen
- Det andet er attributter
- Det tredje er operationer

UML

STRUKTUR DIAGRAMMER

Klasse diagrammer

Klasse navn

Egenskaber

- (+ / - / #)
 - offentlig / privat / beskyttet
- :type

Operationer ()

- (+ / - / #)
 - offentlig / privat / beskyttet
- :return

Musician
<ul style="list-style-type: none">-type:String-firstName:String-lastName:String-bandName:String
<ul style="list-style-type: none">+getBand():String+getName():String+getType():String+setBand()+setType()+setName()

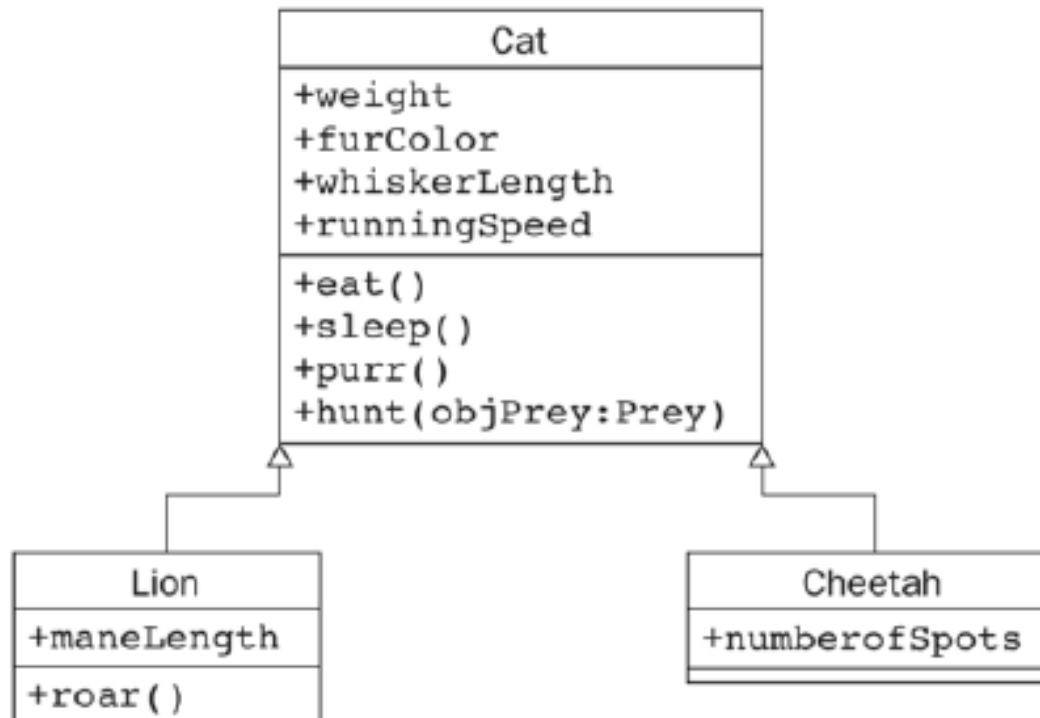
UML

STRUKTUR DIAGRAMMER

Klasse diagrammer

Nedarvning (inheritance)

- En klasse arver fra sin forælder-klasse
- Overskriver værdier i forælderen
- Bibeholde forælderen



UML

STRUKTUR DIAGRAMMER

Klasse diagrammer

En klasse repræsenterer en gruppe af ting, der har fælles tilstand og adfærd

En klasse er repræsenteret af en rektangulær kasse opdelt i rum

- Første rum indeholder navnet på klassen
- Det andet er attributter
- Det tredje er operationer

Et klasse diagram bruges især til at udlægge variabler og

UML

STRUKTUR DIAGRAMMER

Klasse diagrammer og programmering

Klasse diagrammer er de mest populære UML diagrammer brugt i forbindelse med konstruktion af software-applikationer. Så det er meget vigtigt at lære proceduren for klassediagram tegning

Klassediagrammer danner grundlag for overvejelser medens man tegner dem

Et klassediagram er dybest set en grafisk repræsentation af det statiske billede af systemet og repræsenterer forskellige aspekter af anvendelsen. Så en samling af klassediagrammer repræsenterer hele systemet.

Generelt er UML diagrammer er ikke direkte forbundet med eventuelle objektorienterede programmeringssprog men klassediagrammet er en undtagelse

- Klasse diagram viser tydeligt forbindelsen til objektorienterede sprog som Java, C ++ osv.
- Vi kommer senere i forløbet ind på hvordan man direkte kan bruge et klassediagram som grundlag for objekter i C#

UML

STRUKTUR DIAGRAMMER

OPGAVE

Start Astah

Lav et klasse diagram (Class Diagram)

Lav klassen Bil

- Den skal have egenskaberne
 - Producent (bogstav værdi)
 - Model (bogstav værdi)
 - Vægt (tal værdi)
 - Motor (bogstav værdi)
 - Hjul (tal værdi)
- Den skal have operationerne
 - Køre (hastighed) (tal værdi)

UML

STRUKTUR DIAGRAMMER

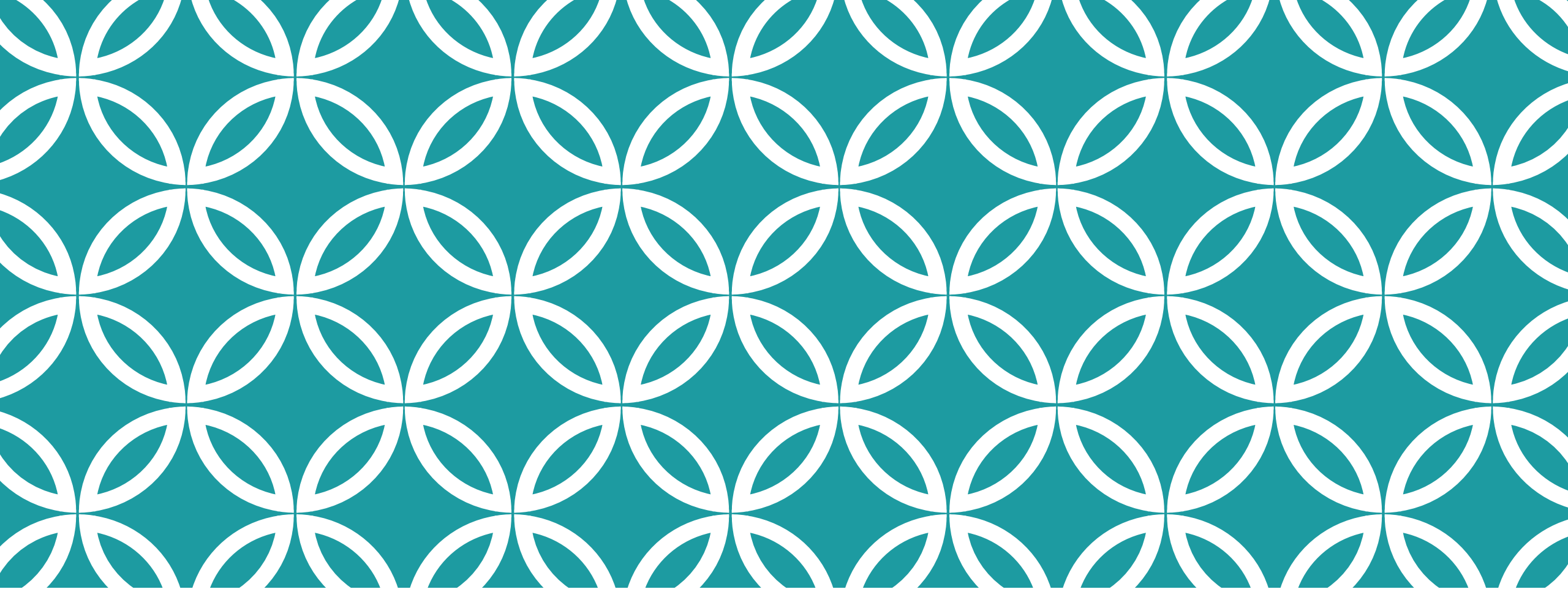
OPGAVE

Lav klassen Personbil

- Den skal være barn af Bil klassen
- Den skal føje egenskaben "Døre (tal værdi)" til
- Den skal føje egenskaben "Antal pladser (tal værdi)" til
- Den skal føje egenskaben "Bagagerum (ja/nej (boolean) værdi)" til
- Den skal føje egenskaben "Trailertræk (ja/nej (boolean) værdi)" til

Lav klassen Varebil

- Den skal være barn af Bil klassen
- Den skal føje egenskaben "Skydedør i siden (ja/nej (boolean) værdi)" til
- Den skal føje egenskaben "Trailertræk (ja/nej (boolean) værdi)" til



OBJEKT ORIENTERET PROGRAMMERING

Struktureret kodning med fokus
på genbrug

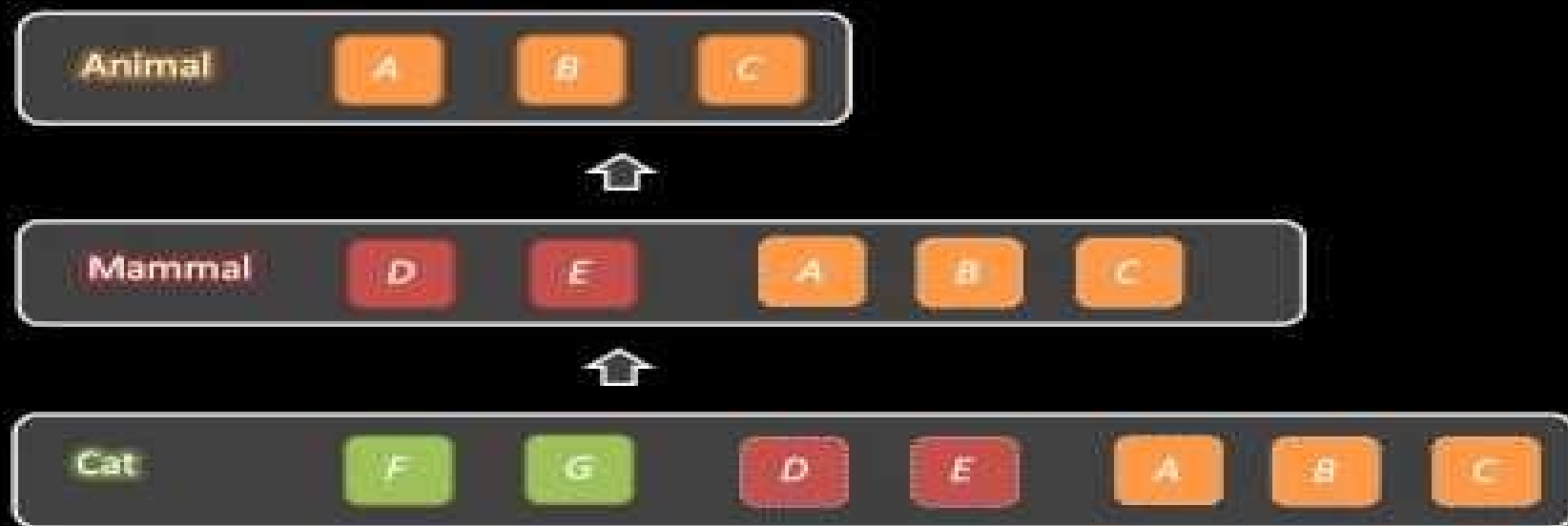
OBJEKT ORIENTERET PROGRAMMERING

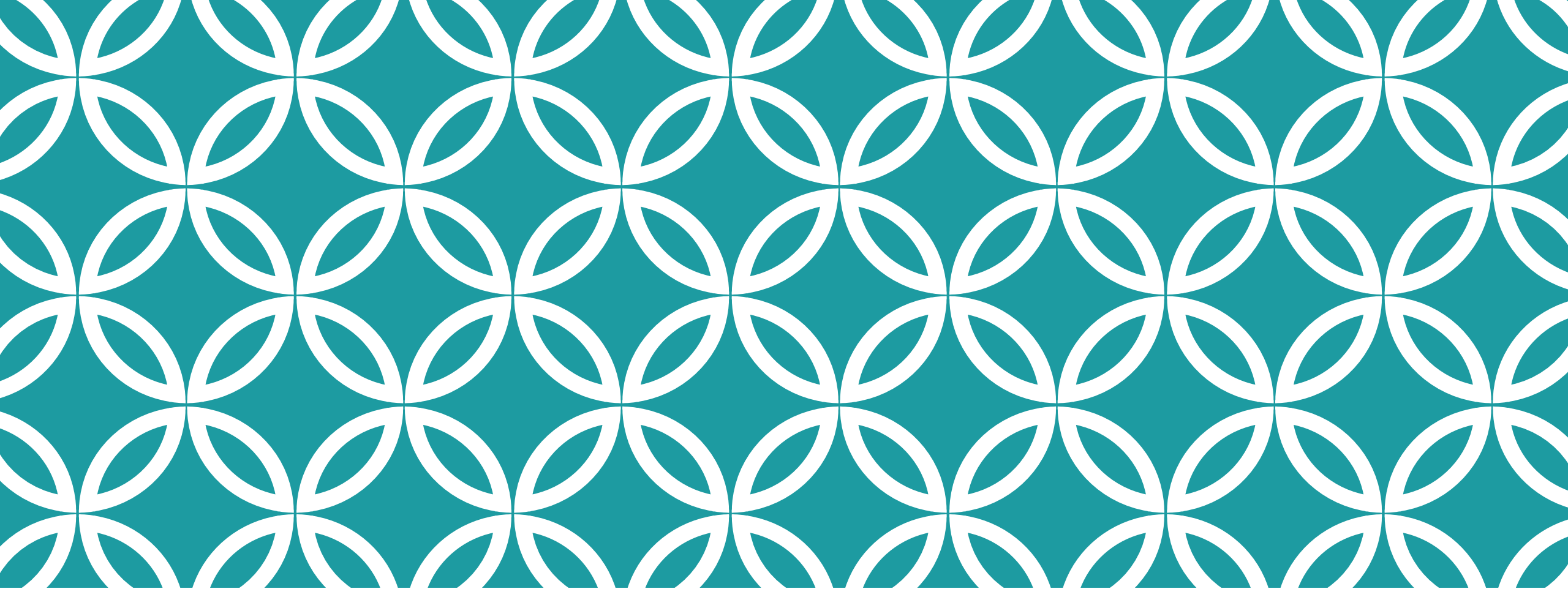
Objekt-orienteret programmering (OOP) henviser til en form for computer programmering (software design), hvor programmører definerer ikke kun datatypen for en datastruktur, men også de typer af operationer (funktioner), der kan anvendes til datastrukturen.

Fokus er altså på objekter, der gør ting.

- Det er altså direkte parallelt med klasse diagrammerne
 - Et objekt er en klasse
 - Objektet har *egenskaber*
 - Det er f.eks. en variabel der indeholder en tal-værdi og en tekst-streng
 - Man kan foretage *operationer* med objektet
- På den måde kan man benytte UML klasse diagrammerne til direkte at forberede ens objekter

OBJEKT ORIENTERET PROGRAMMERING





GRUNDLÆGGENDE BEGREBER

Grundlæggende definitioner i
C# og generel programmering

GRUNDLÆGGENDE BEGREBER

Indhold i et C# program

Et C # program består af følgende dele:

- Namespace deklaration

- En klasse

- Klasse metoder

- Klasse attributter

- En Main metode

- Statements og Expressions

- Kommentarer

GRUNDLÆGGENDE BEGREBER

Indhold i et C# program

Noter omkring C#

C # er case sensitive.

Alle statements og udtryk skal afsluttes med et semikolon (;).

Udførelsen af programmet starter ved Main metoden.

I modsætning til Java må programmets filnavn være forskelligt fra klasse navnet.

GRUNDLÆGGENDE BEGREBER

Indhold i et C# program

En variation over eksemplet vi har brugt hidtil:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace HelloWorldApplication
8  {
9      class HelloWorld
10     {
11         static void Main(string[] args)
12         {
13             /* my first program in C# */
14             Console.WriteLine("Hello World");
15             Console.ReadKey();
16         }
17     }
18 }
```

Behold dette slide åbent medens jeg går videre i slideshowet
fælles så I kan referere til koden

GRUNDLÆGGENDE BEGREBER

Indhold i et C# program

Eksemplet følger strukturen beskrevet tidligere:

De første linjer med **using** viser at programmet inkluderer forskellige præ-definerede *namespaces*.

- Et program benytter typisk flere using statements.
- Hvis vi ikke havde angivet system først skulle vi senere skrive system foran, f.eks. `system.console.writeline`
- Vi kunne have lavet et statisk direktiv og sagt `using.system.console` i toppen hvorefter vi så ikke havde behøvet kalde console foran writeline

Derefter deklareres **namespace**.

- Et namespace er en samling klasser.
- I dette tilfælde klassen *HelloWorld*

Den næste linje er en **klasse** (*class*) deklaration, klassen HelloWorld der indeholder den data og de metode deklarationer vores program benytter.

Metoder definerer hvordan klasserne opfører sig, men HelloWorld klassen har dog kun én metode: Main.

Indhold i et C# program

Eksemplet følger strukturen beskrevet tidligere:

Den næste linje definerer *Main* metoden, der er starten på alle C# programmer. Main metoden angiver hvad klassen gør når udført.

Den næste linje er `/*...*/` ignoreres af compileren da det er **kommentarer**.

Main metodens opførsel angives med statementet **`Console.WriteLine("Hello World");`**

- *WriteLine* er en metode i *Console* klassen defineret i *System* namespace. Dette statement sørger for at beskeden "Hello, World!" vises på skærmen.
- Den sidste linje **`Console.ReadKey();`** er til når vi kører det i VS.NET. Dette får programmet til at vente på at der trykkes en tast og forhindrer dermed skærmen fra at lukke for hurtigt når vi kører programmet fra Visual Studio .NET.

GRUNDLÆGGENDE BEGREBER

Indhold i et C# program

Endnu et eksempel

Visual Studio > New > Project > Visual C# > Windows > Console Application

```
1  using System;
2  namespace RectangleApplication
3  {
4      2 references
5      class Rectangle
6      {
7          // member variables
8          double length;
9          double width;
10         1 reference
11         public void Acceptdetails()
12         {
13             length = 4.5;
14             width = 3.5;
15         }
16         1 reference
17         public double GetArea()
18         {
19             return length * width;
20         }
21     }
22     0 references
23     class ExecuteRectangle
24     {
25         0 references
26         static void Main(string[] args)
27         {
28             Rectangle r = new Rectangle();
29             r.Acceptdetails();
30             r.Display();
31             Console.ReadLine();
32         }
33     }
34 }
35
36
37
38
```

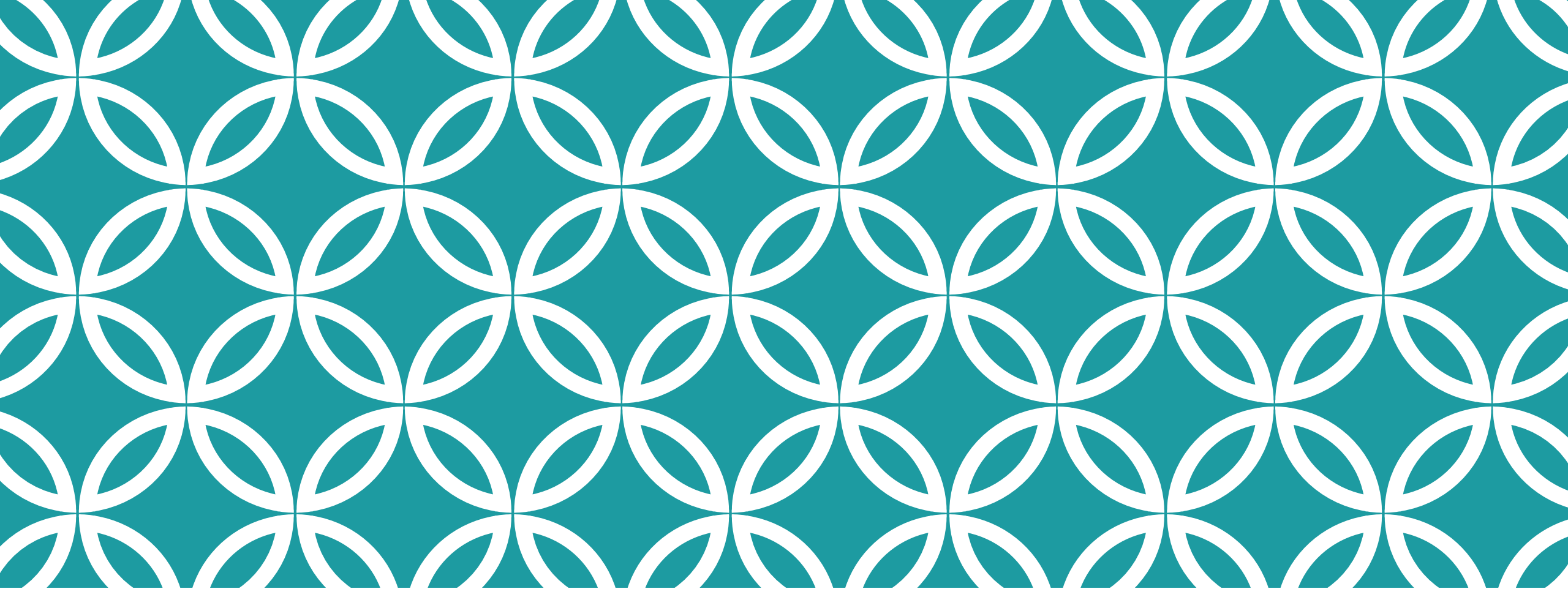
Indhold i et C# program

double angiver en simpel type, der gemmer en 64-bit floating-point-værdi (15-16 cifre).

public når man deklarerer en metode fortæller compilere at metoden skal være bredt tilgængelig for alt i programmet, ikke kun for den klasse den hører under.

void bruges som output type for en metode, hvor det angiver, at metoden ikke returnerer en værdi.

Length: {0} angiver at vi vil have første værdi i length, da det kunne have været et array mere flere værdier og dermed flere pladser.

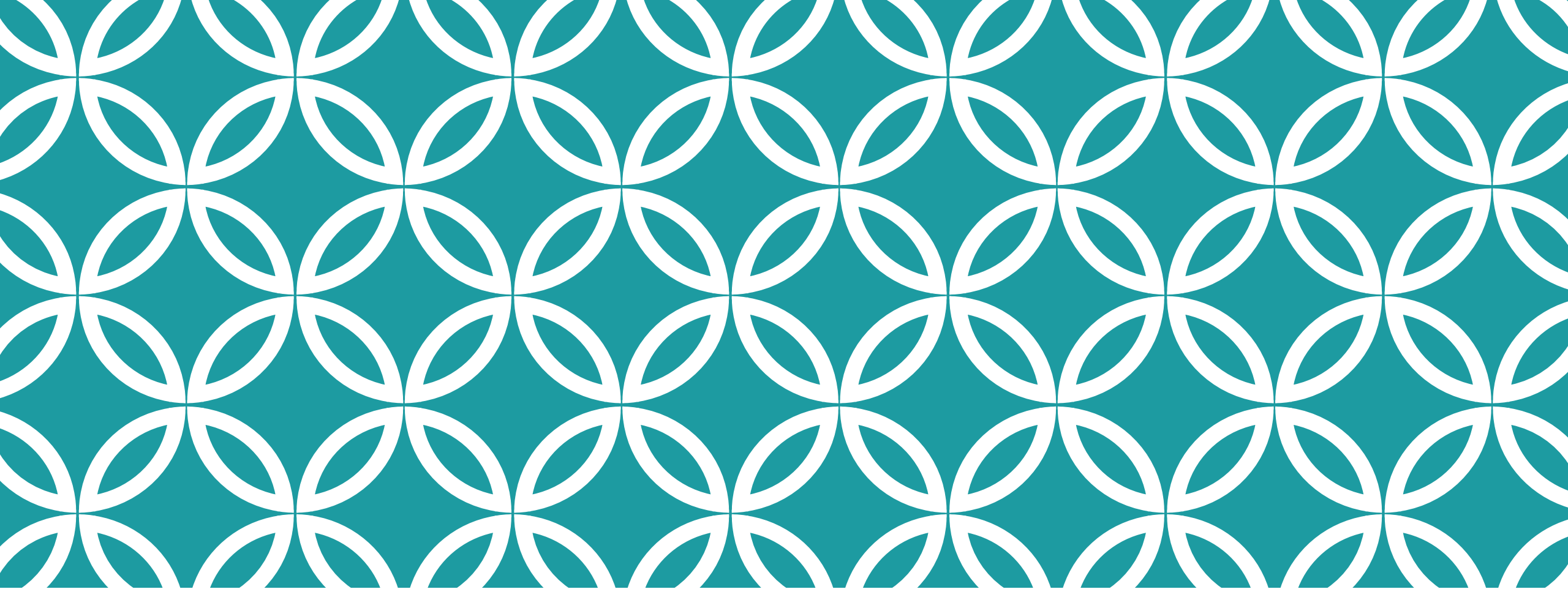


C# NØGLEORD

Reservede ord

C# NØGLEORD

Reserved Keywords						
abstract	as	base	bool	break	byte	case
catch	char	checked	class	const	continue	decimal
default	delegate	do	double	else	enum	event
explicit	extern	false	finally	fixed	float	for
foreach	goto	if	implicit	in	in (generic modifier)	int
interface	internal	is	lock	long	namespace	new
null	object	operator	out	out (generic modifier)	override	params
private	protected	public	readonly	ref	return	sbyte
sealed	short	sizeof	stackalloc	static	string	struct
switch	this	throw	true	try	typeof	uint
ulong	unchecked	unsafe	ushort	using	virtual	void
volatile	while					
Contextual Keywords						
add	alias	ascending	descending	dynamic	from	get
global	group	into	join	let	orderby	partial (type)
partial (method)	remove	select	set			



MANGE SPROG

Der findes rigtig, rigtig mange programmeringssprog
De følgende er kun nogle få eksempler

MANGE SPROG

Bash

Er det “sprog” man bruger i konsollen på Unix (herunder også BSD som MacOS) og Linux maskiner, men i og med at man kan bede OS om at lave en række handlinger kan man kalde det programmering.

```
echo "Hello World"
```

MANGE SPROG

Basic

Sproget som mange begyndte med i 80'erne og 90'erne. Som navnet siger var tanken bag at det skulle være meget simpelt, men det var så på bekostning af mere avancerede muligheder og effektivitet.

```
PRINT "Hello, world!"
```

MANGE SPROG

C

“Moderen” i C-familien af sprog, et af de mere effektive men også komplicerede høj-niveau sprog men der findes C-compile til alle systemer i dag så kan man C skal man “blot” sætte sig ind i de enkelte platformes unikke biblioteker.

```
#include
```

```
int main(void)
```

```
{
```

```
    puts("Hello, world!");
```

```
}
```

MANGE SPROG

C++

En videreudvikling af C som i dag er det dominerende af de klassiske C sprog.

```
#include  
int main()  
{  
    std::cout << "Hello, world!"  
    ;  
    return 0;  
}
```

MANGE SPROG

C#

Det vi fokuserer på i dette kursus

```
using System;  
class Program  
{  
    public static void Main(string[] args)  
    {  
        Console.WriteLine("Hello, world!");  
    }  
}
```

MANGE SPROG

HTML

HTML er det scriptsprog man opbygger hjemmesider i, så tekst man bare skriver i et HTML dokument vises som tekst på skærmen.

Hello World!

MANGE SPROG

Java

Java er på trods af navnet også et sprog i C-familien, da det er inspireret af C's opbygning og måder at gøre ting på.

```
import javax.swing.JFrame; //Importing class JFrame
import javax.swing.JLabel; //Importing class JLabel
public class HelloWorld {
    public static void main(String[] args) {
        JFrame frame = new JFrame();           //Creating frame
        frame.setTitle("Hi!");                  //Setting title frame
        frame.add(new JLabel("Hello, world!")); //Adding text to frame
        frame.pack();                           //Setting size to smallest
        frame.setLocationRelativeTo(null);      //Centering frame
        frame.setVisible(true);                 //Showing frame
    }
}
```

MANGE SPROG

JavaScript

JS er *ikke* en afart af Java men et selvstændigt script sprog til at understøtte HTML i at lave dynamiske og interaktive sider til internettet, det kan dog bruges også til at lave selvstændige apps med de rette værktøjer.

```
document.write('Hello, world!');
```


MANGE SPROG

jQuery

jQuery er egentlig et JavaScript bibliotek der gør en masse funktioner lettere tilgængelige ved at præ-definere dem. Man ser det dog ofte nævnt på listen over sprog som firmaer søger - men det er ikke et selvstændigt sprog, bare en udvidelse!

```
$("body").append("Hello world!");
```

MANGE SPROG

Matlab

MatLab er et værktøj til at lave komplicerede beregninger med der er ekstremt elsket i universitets-miljøer, og på dette niveau af kompleksitet kan det selvfølgelig også programmeres.

```
disp('Hello, world!')
```

MANGE SPROG

Objective-C

Apples C-sprog rettet mod at lave programmer til deres udstyr.

```
#import  
int main(void)  
{  
    NSLog(@"Hello, world!  
");  
    return 0;  
}
```

MANGE SPROG

Pascal

Endnu et sprog tilbage fra 90'erne som mange lærte, typisk som det første efter Basic

```
program HelloWorld;
```

```
begin
```

```
    WriteLn('Hello, world!');
```

```
end.
```

MANGE SPROG

Python

Python fokuserer på læsbarhed og på at kunne opnå det samme som C++ og Java med færre linjer.

```
print "Hello, world!"
```

MANGE SPROG

Perl

Perl blev skabt under Unix for at gøre rapportering lettere. Siden 1987 har Perl undergået store forandringer og er i dag et generelt anvendeligt programmeringssprog, som kan afvikles på flere forskellige platforme, herunder bl.a. Microsoft Windows og Unix.

```
print "Hello, world!
```

```
";
```

MANGE SPROG

Ruby

Et dynamisk open source programmerings sprog der lægger vægt på enkelhed.

```
puts "Hello, world!"
```

MANGE SPROG

Swift

Apples ny sprog med fokus på at lave iOS, macOS, watchOS, og tvOS apps.

```
println("Hello, world!")
```


MANGE SPROG

VBScript

VBScript er et "script"-sprog, der fortolkes via Microsoft's Windows Script Host. Sprogets syntax viser dets oprindelse som en variation af Microsoft's Visual Basic programmeringssprog.

```
MsgBox "Hello, World!"
```

MANGE SPROG

Visual Basic .NET

Efterfølger til programmeringssproget Visual Basic. Visual Basic .NET er et fuldgyldigt sprog på .NET platformen, men er dog ikke lige så stort som C#. Visual Basic og Visual Basic .NET, ligner hinanden en smule, men der er dog en del forskelle, og sprogene kan ikke blandes.

```
Module Module1
```

```
    Sub Main()
```

```
        Console.WriteLine("Hello, world!")
```

```
    End Sub
```

```
End Module
```

MANGE SPROG

SQL

SQL er det sprog man kommunikerer til database-servere med, og som sådan kan man derfor ikke bede det om at skrive “Hello World” til skærmen.

MANGE SPROG

De sjove sprog

Programmører har humor, den er bare ofte lidt anderledes og meget mere indforstået end "den almindelige".

Således findes der også en række sprog der enten er lavet fordi opfinderen ville give andre en udfordring ellers simpelthen ville lave noget der var så underligt at det var komisk i sig selv. Som et underholdende indslag bringer jeg tre af dem – men det er ikke noget der bliver brugt seriøst!

MANGE SPROG DE SJOVE SPROG

ArnoldC

Sproget består udelukkende af one-liners fra Arnold Schwarzenegger film. Sproget blev lavet af Lauri Hartikka, der udskiftede normale kommandoer med Arnold citater. False og True er f.eks. "I lied" og "No problemo".

IT'S SHOWTIME

TALK TO THE HAND "Hello World!"

YOU HAVE BEEN TERMINATED

MANGE SPROG DE SJOVE SPROG

Brainfuck

Lavet i 1993 af Urban Müller som et lille sprog der skulle underholde programmører. Sproget indeholder kun otte kommandoer og en instruktions pointer. De består hver af ét tegn, så sproget er ekstremt minimalistisk.

```
+++++++[[>++++++>+++++++>+++>+<<<<-  
]>++.>+.+++++..+++.>+<<+++++++>+.+++.-----  
.>+.>.
```

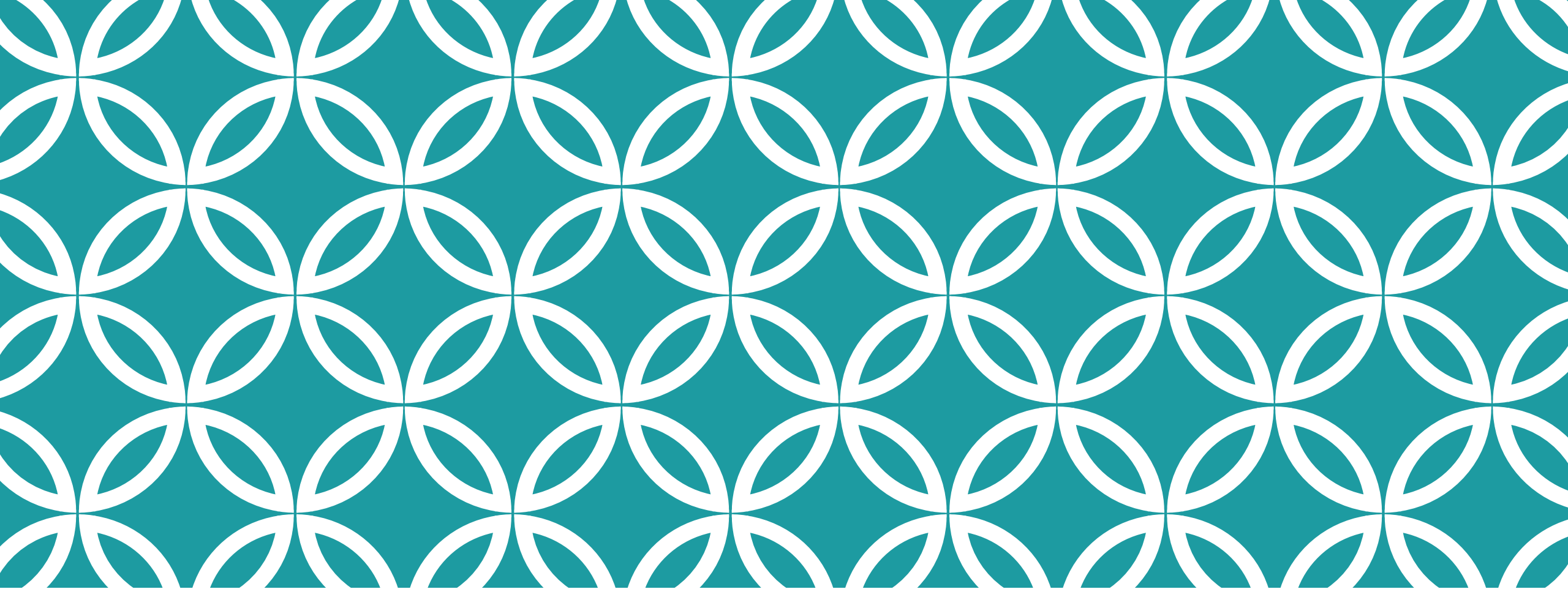
MANGE SPROG DE SJØVE SPROG

Piet

Sproget er opkaldt efter den hollandske maler Piet Mondrian, der malede billeder bestående af firkanter. Det er også hvad programmerings sproget gør, da det kompileres af en pointer der kører gennem et billede, fra et området til det næste, og laver programmet ud fra farverne.

"Hello World!" ser således sådan ud:





LEKTIE

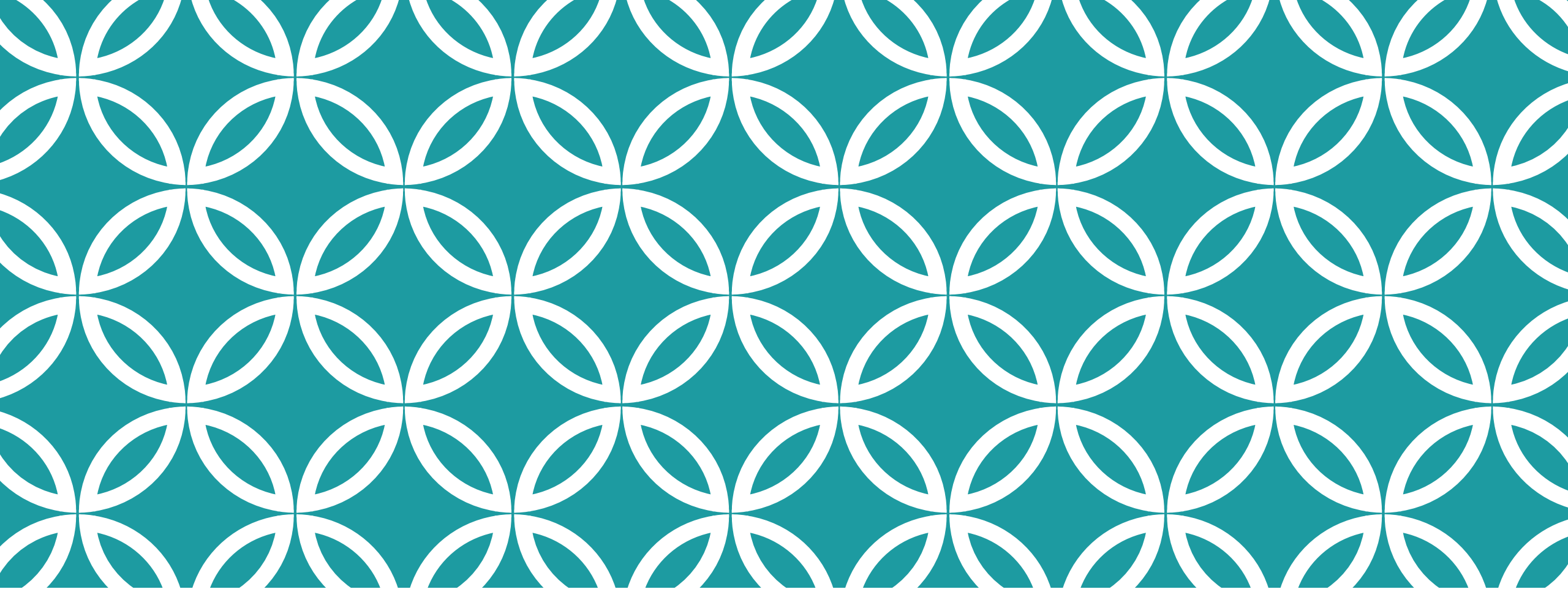
Kig på dette til næste gang

LEKTIE

Se video 4-5 i <https://mva.microsoft.com/en-US/training-courses/c-fundamentals-for-absolute-beginners-16169>

https://www.tutorialspoint.com/csharp/csharp_overview.htm til

https://www.tutorialspoint.com/csharp/csharp_type_conversion.htm



KILDER

Materiale benyttet i denne
lektion
Noget af det er udover pensum-
listen!

KILDER

UML

https://www.tutorialspoint.com/uml/uml_class_diagram.htm

<https://msdn.microsoft.com/en-us/library/dd409416.aspx>

<http://creately.com/blog/diagrams/umldiagram-types-examples/>

<http://modeling-languages.com/best-uml-cheatsheets-and-reference-guides/>

Objekt Orienteret Programmering

http://www.webopedia.com/TERM/O/object_oriented_programming_OOP.html

<https://youtu.be/lbXsrHGhBAU>

Grundlæggende begreber

<https://www.tutorialspoint.com/csharp/index.htm>

<https://msdn.microsoft.com/da-dk/library/678hzkk9.aspx> (double)

<https://msdn.microsoft.com/en-us/library/yzh058ae.aspx> (public)

KILDER

Mange sprog

<https://excelwithbusiness.com/blog/post/web-design/say-hello-world-in-28-different-programming-languages>

<http://www.hongkiat.com/blog/bizarre-insane-programming-languages/>

[https://esolangs.org/wiki/Hello world program in esoteric languages](https://esolangs.org/wiki/Hello_world_program_in_esoteric_languages)