2

Samenvatting

In dit hoofdstuk komen de codeconventies van zowel C# als php aanbod

Code conventies

Gemaakt door groep 3

Inhouds

[1. PHP conventies 2](#_Toc483743508)

[1.1 Globale conventies 2](#_Toc483743509)

[1.2 Samenvatting conventies 2](#_Toc483743510)

[1.3 Layout conventies 2](#_Toc483743511)

[1.4 Variabele conventies 2](#_Toc483743512)

[1.5 Methods conventies 2](#_Toc483743513)

[1.6 Functies conventies 2](#_Toc483743514)

[1.7 Loop en If-statements 2](#_Toc483743515)

[1.8 Classes 3](#_Toc483743516)

[2. C# conventies 3](#_Toc483743517)

[2.1 globale conventies 3](#_Toc483743518)

[2.2 Samenvatting conventies 3](#_Toc483743519)

[2.3 Layout conventies 3](#_Toc483743520)

[2.4 variabele conventies en 2.5 Methodes conventies 3](#_Toc483743521)

[2.6 functies conventies 3](#_Toc483743522)

[2.7 Loop en If-statements en 2.8 classes 3](#_Toc483743523)

# 1. PHP conventies

In dit hoofdstuk komen de PHP conventies aanbod.

## 1.1 Globale conventies

1. Code schrijven in 1 taal (Engels).
2. Het goed gebruiken van code inspringen (meerdere lijnen).
3. Het goed toepassen van pascal en camel case.
4. Code altijd beginnen met <? PHP.
5. De code altijd afsluiten met?>

## 1.2 Samenvatting conventies

1. Alle kleine commentaar worden met // aangegeven.
2. Alle grotere commentaar worden met ?/\*\*/ aangegeven.

## 1.3 Layout conventies

1. Elke nieuwe code krijgt een eigen regel.
2. Voor elke nieuwe class, statement, loops, methodes, functies enz. komt een witte regel. Dit maakt het lees vriendelijk voor de mee werkende collega’s.
3. Tussen elke div’s komen een witte regel voor.

## 1.4 Variabele conventies

1. Alle variabel beginnen met den Dollar taken: $. Dit is een globale afspraak in deze programmeertaal.
2. De schrijfwijzen is als volgt.
   1. Eest deel is camel case. Voorbeeld = $test
   2. De spaties zijn met een liggend streepje. Voorbeeld = $test\_test
   3. Alles na een spatie begint met een hoofdletter. Voorbeeld: = $test\_Test
   4. Arrays worden altijd in meervoud geschreven.

## 1.5 Methods conventies

1. De benaming is altijd lower Cased.
2. Alle methodes zijn in werkwoord wijzen gespeld.
3. Brackets komen altijd in de volgende regel te staan.
4. Alle methodes komen onder aan de code te staan.

## 1.6 Functies conventies

1. De benaming is altijd lower Cased.
2. Alle methodes zijn in werkwoord wijzen gespeld.
3. Brackets komen altijd in de volgende regel te staan.
4. Alle functies komen boven de methodes te staan.

## 1.7 Loop en If-statements

1. Alle loops en statement worden op de originele schrijf wijze van PHP.
2. Brackets Komen altijd op de volgende regel.
3. Code begint met een lege regel.
4. Voor de if statements en for loops worden kleine uitleg meegeven. Dit is niet genoodzaakt een elke taak uit voor in de programma.
5. Foreach loops worden een variabele aan gemaakt waarvan de naam in enkelvoud van de array.

## 1.8 Classes

1. Underscores gebruiken als spaties in de naam van classes.
2. Brackets altijd op een nieuwe regel
3. Eerste regel begint met een tegel.

# 2. C# conventies

In dit hoofdstuk komen de C# conventies aanbod

## 2.1 globale conventies

1. De code schrijven we in het Engels.
2. Bestandsnaam in lowercase.
3. Gebruik gedefinieerde type names inplaats van systeem types.

## 2.2 Samenvatting conventies

1. Alle kleine commentaar worden met // aangegeven.
2. Alle grotere commentaar worden met ?/\*\*/ aangegeven.
3. Standaard comments gebruiken als uitleg wat de loop doet.

## 2.3 Layout conventies

1. Tussen de Methodes een lege balk.
2. Hoofdletter gevoelig dus geen hoofdletters

## 2.4 variabele conventies en 2.5 Methodes conventies

1. Camelcase voor methode argumenten en lokale variabelen.
2. Enkele quotes gebruiken maar als het niet anders kan dubbele quotes gebruiken

## 2.6 functies conventies

1. PascalCase altijd eenvoud en een eigen naamwoord.

## 2.7 Loop en If-statements en 2.8 classes

1. Pascalcase voor class names en methode names.
2. Underscores gebruiken als spaties in de naam van classes.