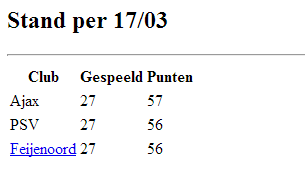
# Terminal toets Blok C

## Onderwijseenheid bouwen

### Vraag 1 (HTML en CSS)

a. **(1 pnt)**

Zet onderstaande afbeelding exact over in html. Je moet minimaal gebruik maken van 1 tabel. Zorg ervoor dat wanneer de gebruiker op Feijenoord klikt hij wordt doorverwezen naar <http://www.feijenooord.nl>.



b. **(1 pnt)**

Zet onderstaande afbeelding exact over in html en **css**. Je mag alleen maar gebruik maken van **div** tags en de opmaak moet in zijn **geheel** in **css** zijn opgenomen. De breedte van de afbeelding is 400px en de hoogte 300 px. Verder zijn de volgende kleuren gebruikt **#cc0000**, **#c0c0c0** en **#000000. (1pnt)**



### Vraag 2 (C# en ADO.NET)

**Open Visual Studio project Toetsvraag2 en maak de database toetsvraag2 aan. Gebruik hiervoor het meegeleverde bestand toetsvraag2.sql**

a.**(1 pnt)**

Schrijf de klasse **MobieleTelefoon**. Deze klasse heeft de properties **ID(int)**, **Fabrikant(string),** **Type(string)**, **Prijs(double)** en **OperatingSystem(OperatingSystem)**. **(1 pnt)**

b. **(1 pnt)**

Geef een implementatie van **de ToString()** methode van de klasse **MobieleTelefoon**. Hieronder een voorbeeld van wat deze methode zou moeten retourneren.

***Fabrikant = Samsung***

***Type = Galaxy 10***

***Prijs = 350***

***Operating system = (Naam = Android, Versie = 1.0.1)***

**c. (1 pnt)**

Test de methode die je bij opgave b geschreven hebt door een object aan te maken van het type **MobieleTelefoon** en de textuele representatie van dit object naar de Console te schrijven.

d. **(1 pnt)**

Implementeer de methode **AddOperatingSystem** uit de klasse **OperatingSystemDBController.** Deze methode moet ervoor zorgen dat het **OperatingSystem,**  dat als parameter wordt meegeven, wordt opgeslagen in de database. Test je methode door een object aan te maken van het type **OperatingSystem** en deze op te slaan in de database door de door jouw geschreven methode **AddOperatingSystem(OperatingSystem oper)** aan te roepen.

### Vraag 3 (ASP.NET MVC3 )

1. **(1 pnt)**

Maak een nieuw ASP.NET MVC 3 project aan. Implementeer binnen dit project de juiste Controller en View, zodat het onderstaande zichtbaar wordt. Let op je hoeft nog geen implementatie te schrijven achter de bereken knop.



b. **(1 pnt)**

De input moet op onderstaande manier gevalideerd worden, zorg voor duidelijke foutmeldingen voor de gebruiker. **(indien eerste twee bullets dan 0.5 pnt)**

* **Minimale waarde** is een **verplicht veld** en moet **tussen de 1 en de 100 liggen.**
* **Maximale waarde** is een **verplicht veld** en moet **tussen de 2 en de 200 liggen.**
* De **minimale waarde** mag **niet groter** zijn **dan** de **maximale waarde.**

**c. (1 pnt)**

Maak de applicatie nu werkend. Op het moment dat de gebruiker op de knop drukt moeten de tafels van de minimaal ingevulde waarde tot en met de maximaal ingevulde waarde berekend worden en op het scherm worden getoond. Dit moet natuurlijk alleen gebeuren wanneer de ingevoerde waarden valide zijn. Hieronder zie je een voorbeeld van de output wanneer een gebruiker 6 en 7 heeft ingevoerd.

