# Computing, Year 3

# Rich Web Applications

**Lab 3**

# Getting Started with TypeScript Syntax

Get familiar with TypeScript : ***http://www.typescriptlang.org***

## Overview

In this lab, you'll define a class in TypeScript to hold information and behaviour for a film. You'll also write some simple TypeScript client code to create film object(s) and invoke their methods.

## Roadmap

There are 2 exercises in this lab. Here is a brief summary of the tasks you will perform in each exercise; more detailed instructions follow later:

1. Defining a simple film class
2. Implementing film behavior

## Exercise 1: Defining a simple film class

Open a browser and go to http://www.typescriptlang.org/play. This is the online TypeScript Language Playground web site, which makes it easy for you to enter TypeScript code and benefit from syntax checking as you go.

In the Playground window, define a Film class to hold the following details for a film (make sure you define these fields as private for encapsulation purposes, and make full use of TypeScript's support for string data typing):

* A numeric id (each film should have a unique id – how will you achieve this?)
* A title
* Blurb about the film (you know, the kind of info you get on the back of the DVD case)
* The price of the film
* A collection of genres for the film (e.g. the genres for Star Wars might be 'adventure' and 'science fiction')
* A collection of score ratings for the film (the idea here is that users can rate the film between 1 and 5 inclusive)

Define a suitable constructor to initialize these fields as appropriate. Then write a small piece of client code to create a new Film object. Verify there are no syntax errors before you proceed.

## Exercise 2: Implementing film behaviour

Add some methods to the Film class:

* addGenres()  
  This method should be variadic – it should receive any number of strings and add them all to the film's collections of genres.
* genresAsString()  
  This method should return a string concatenation of all the genres for the film.
* rate()  
  This method should receive a score rating between 1 and 5 inclusive, and add it to the film's collections of scores.
* averageScore()  
  This method should calculate and return the average score rating for the film.
* toString()  
  This method should return a string representation of all the film's data. You choose the format of this string.

Write client code to invoke the methods you've just implement. Call console.log() to display information on the browser console window. Then run the code and view the browser's Console window, to verify the output is correct.