<https://github.com/rvsp/typescript-oops/blob/master/Practice/Movie.md>

# **Class - Movie**

The class Movie is stated below. An instance of class Movie represents a film. This class has the following three properties:

* title, which is a String representing the title of the movie
* studio, which is a String representing the studio that made the movie
* rating, which is a String representing the rating of the movie (i.e. PG­13, R, etc)

a) Write a constructor for the class Movie, which takes a String representing the title of the movie, a String representing the studio, and a String representing the rating as its arguments, and sets the respective class properties to these values.

b) The constructor for the class Movie will set the class property rating to "PG" as default when no rating is provided.

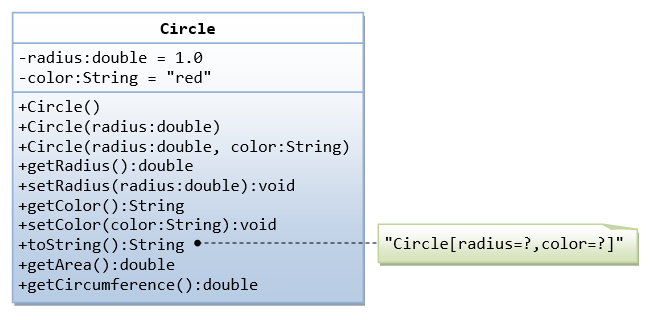
c) Write a method getPG, which takes an array of base type Movie as its argument, and returns a new array of only those movies in the input array with a rating of "PG". You may assume the input array is full of Movie instances. The returned array need not be full.

d) Write a piece of code that creates an instance of the class Movie with the title “Casino Royale”, the studio “Eon Productions”, and the rating “PG­13”

<https://github.com/rvsp/typescript-oops/blob/master/Practice/class-circle.md>

# **Circle - Class**

Convert the UML diagram to Typescript class. - use number for double

[](https://github.com/rvsp/typescript-oops/blob/master/images/ClassDiagram_Circle.png)

Write a “person” class to hold all the details.

function Person() {

this.firstName = "unknown";

this.lastName = "unknown";

}

var person1 = new Person();

person1.firstName = "Steve";

person1.lastName = "Jobs";

alert(person1.firstName + " " + person1.lastName);

var person2 = new Person();

person2.firstName = "Bill";

person2.lastName = "Gates";

alert(person2.firstName + " " + person2.lastName );

write a class to calculate uber price.

I edited the whole topic because I gave a great read about the whole API and now I'm having another problem ... I'll try to leave as much details as possible here.

The error you are giving is 401 of uber api, but, I did everything right ... My INDEX.html looks like this:

***index.html***

<html>

<head>

<title>Party Invitation</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<link href='http://fonts.googleapis.com/css?family=Source+Sans+Pro:200,300,400,600' rel='stylesheet' type='text/css'>

<link rel="stylesheet" href="css/normalize.css">

<link rel="stylesheet" href="css/style.css">

</head>

<body>

<section>

<h1>Party</h1>

<div class="info-container">

<span class="info-header">When</span>

<span class="info when">September 18, 2014 at 6:00pm</span>

</div>

<div class="info-container">

<span class="info-header">Where</span>

<span class="info where">Thinkful HQ</span>

</div>

<img class="map" src="http://maps.googleapis.com/maps/api/staticmap?center=40.7248,-73.99597&zoom=17&format=png&sensor=false&size=280x280&maptype=roadmap&style=element:geometry.fill|color:0xf4f4f4&markers=color:red|40.7248,-73.99597&scale=2" alt="">

<div class="button">

<p id="time">ESTIMATING TIME</p>

</div>

</section>

<script src="js/jquery-2.1.1.min.js"></script>

<script src="js/uber.js"></script>

</body>

And then my uber.js is like this:

***uber.js***

// Uber API Constants

var uberClientId = "RxGbzH\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

, uberServerToken = "h\_hqd3L4\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

// Create variables to store latitude and longitude

var userLatitude

, userLongitude

, partyLatitude = 19.3256725

, partyLongitude = -43.1579731;

navigator.geolocation.watchPosition(function(position) {

// Update latitude and longitude

userLatitude = position.coords.latitude;

userLongitude = position.coords.longitude;

// Query Uber API if needed

getEstimatesForUserLocation(userLatitude, userLongitude);

});

function getEstimatesForUserLocation(latitude,longitude) {

$.ajax({

url: "https://api.uber.com/v1/estimates/price",

headers: {

Authorization: "KA.ey\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" + uberServerToken

},

data: {

start\_latitude: latitude,

start\_longitude: longitude,

end\_latitude: partyLatitude,

end\_longitude: partyLongitude

},

success: function(result) {