# Department of Computing

**SE-312: Software Construction**

**Class: BESE 9AB**

# Lab 03: Intro to ES6

**Date: 8th March 2021**

**Hassaan Ali Mehmood**

**BESE 9A  
264947**

**Time: 09:00-11:50pm & 02:00-04:50pm**

# Instructor: Dr. Seema Jehan

**Lab Engineer: Mr. Aftab Farooq**

# 

# Lab 03: Intro to ES6

**Objectives**

The objective of this lab is helping students to familiarize themselves with basic concepts of the ES6 constructs. They will practice the concept of classes, subclasses, template strings, default parameters, maps, arrow functions and destructuring.

**Tools/Software Requirement**

Notepad, browser

**Helping Material:**

File Uploaded on LMS

**Lab Tasks:**

**TASK1:**

Suppose that you're working in a small town administration, and you're in charge of two town elements:

1. Parks

2. Streets

It's a very small town, so right now there are only 3 parks and 4 streets. All parks and streets have a name and a build year.

At an end-of-year meeting, your boss wants a final report with the following:

1. Tree density of each park in the town (formula: number of trees/park area)

2. Average age of each town's park (formula: sum of all ages/number of parks)

3. The name of the park that has more than 1000 trees

4. Total and average length of the town's streets

5. Size classification of all streets: tiny/small/normal/big/huge. If the size is unknown, the default is normal

All the report data should be printed to the console.

HINT: Use some of the ES6 features: classes, subclasses, template strings, default parameters, maps, arrow functions, destructuring, etc.

|  |
| --- |
| **Solution** |
| **Task 1 Code:**  class NameAndYear{      constructor(name,year){  this.name=name;  this.year=year;  }  }  class Park extends NameAndYear {  constructor(name, year, NoOfTrees, ParkArea) {  super(name, year);  this.NoOfTrees = NoOfTrees;  this.ParkArea = ParkArea;    }    ParkDensity(){    return this.NoOfTrees/this.ParkArea;  }      AgeofPark(){    let Age= new Date().getFullYear() - this.year;  return Age;  }    }  let Park1 = new Park('Park1', 2005, 200, 0.3);let Park2 = new Park('Park2', 2010, 1200, 1.5);let Park3 = new Park('Park3', 2015, 900, 3.8);    let ParkArray = [Park1, Park2, Park3];      function AverageAgeofPark(){    var TotalAgeOfPark=0;  ParkArray.forEach((data) => {    TotalAgeOfPark+=data.AgeofPark();    });    console.log(`Average age of Park is ${Math.round(TotalAgeOfPark / 3)} years`);    }    function DesnityofPark() {  for (let i of ParkArray) {  console.log(`The Tree Density of ${i.name} is ${Math.round(i.ParkDensity())} per square km`);  }  }    function ParkName(){    ParkArray.forEach((data) => {    if (data.NoOfTrees >= 1000) {  console.log(`${data.name}: has a ${data.NoOfTrees} trees`);  }  });    }  console.log("--------PARK REPORT--------")  DesnityofPark();  AverageAgeofPark();  ParkName();      // STREET  class Street extends NameAndYear {  constructor(name, year, StreetLength, Size = 'normal') {  super(name, year);  this.StreetLength = StreetLength;  this.Size = Size;    }    }  const STREET1 = new Street('STREET1', 2000, 300);const STREET2 = new Street('STREET2', 2005, 1300, 'huge');const STREET3 = new Street('STREET3', 2010, 1100, 'big');const STREET4 = new Street('STREET4', 2015, 100, 'small');    let StreetArray = [STREET1, STREET2, STREET3, STREET4];    function TotalAndAverage() {  let TotalLength = 0;  StreetArray.forEach((data) => {    TotalLength+=data.StreetLength;    });    console.log(`The total length of the town's streets is ${TotalLength} meters and the average length is ${TotalLength / 4} meters`);  }      function SizeClassification(){    StreetArray.forEach((data) => {  console.log(`The size of ${data.name} is ${data.Size}`);  });    }console.log("--------STREET REPORT--------")  TotalAndAverage();  SizeClassification();  **Task 1 Output Screenshot:** |

### Deliverables

Compile a single word document by filling in the solution part and submit this Word file on LMS. This lab grading policy is as follows: The lab is graded between 0 to 10 marks. The submitted solution can get a maximum of 5 marks. At the end of each lab or in the next lab, there will be a viva related to the tasks. The viva has a weightage of 5 marks. Insert the solution/answer in this document. You must show the implementation of the tasks in the designing tool, along with your complete Word document to get your work graded. You must also submit this Word document on the LMS. In case of any problems with submissions on LMS, submit your Lab assignments by emailing it to Mr. Aftab Farooq: [aftab.farooq@seecs.edu.pk](mailto:aftab.farooq@seecs.edu.pk).