Handson 1

Emp.ts

export interface Employee {

id:number;

name:string;

salary:number;

permanent:boolean;

}

Employee.ts

import {Employee} from './emp';

var emp:Employee = {

id:3,

name:"John",

salary:10000,

permanent:true,

}

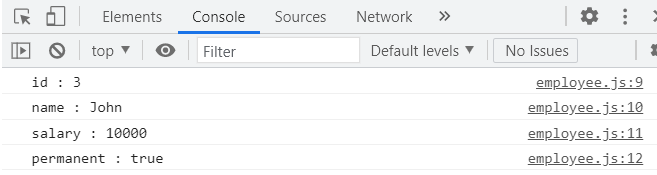
console.log("id : "+emp.id);

console.log("name : "+emp.name);

console.log("salary : "+emp.salary);

console.log("permanent : "+emp.permanent);

Output



Handson-2

Department.ts

export interface Department{

    id:number;

    name:string;

    }

Emp.ts

import { Department } from './Department';

export class Employee implements Department{

id:number;

name:string;

salary:number;

permanent:boolean;

constructor(id:number, name:string, salary:number, permanent:boolean){

this.id = id;

this.name = name;

this.salary = salary;

this.permanent = permanent;

}

}

Employee.ts

import { Department } from './Department';

import { Employee } from './emp';

var emp1 = new Employee(3, "John", 10000, true);

console.log("id: "+emp1.id);

console.log("name: "+emp1.name);

console.log("salary: "+emp1.salary);

console.log("permanent: "+emp1.permanent);

var dept:Department = {

id:2,

name:"payroll",

}

console.log("Department id : "+dept.id);

console.log("Department name: "+dept.name);

Output

