Question 1:

module.exports.lookupByProjectId = function(id){  
 let r = data.filter(object => {  
 return object.projectId == id;  
 });  
 if(r.length){  
 return r[0];  
 }else{  
 return undefined;  
 }  
};

module.exports.lookupByEmployeeId = function(id){  
 let r = data.filter(object => {  
 for(let i = 0; i < object.employees.length;i++){  
 if (object.employees[i].employeeId === id) {  
  
 return true;  
 }  
 }  
 return false;  
 });  
 if(r.length){  
 return ***JSON***.stringify(r, null,2);  
  
 }else{  
 return undefined;  
 }  
};

let data = [  
 {  
 projectName: "Project1",  
 projectId: 'p1',  
 employees: [  
 {fullName: "John Doe", employeeId: 'e1'},  
 {fullName: "John Smith", employeeId: 'e2'}  
 ]  
 },  
  
 {  
 projectName: "Project2",  
 projectId: 'p2',  
 employees: [{fullName: "John Doe", employeeId: 'e1'},  
 {fullName: "Mary John", employeeId: 'e3'},  
 {fullName: "Bill Evan", employeeId: 'e4'}  
 ]  
 }  
];  
  
module.exports.lookupByProjectId = function(id){  
 let r = data.filter(object => {  
 return object.projectId == id;  
 });  
 if(r.length){  
 return r[0];  
 }else{  
 return undefined;  
 }  
};  
  
module.exports.lookupByEmployeeId = function(id){  
 let r = data.filter(object => {  
 for(let i = 0; i < object.employees.length;i++){  
 if (object.employees[i].employeeId === id) {  
  
 return true;  
 }  
 }  
 return false;  
 });  
 if(r.length){  
 return ***JSON***.stringify(r, null,2);  
  
 }else{  
 return undefined;  
 }  
};

Question 2)

**let** data = [  
 {  
 projectName: "Project1",  
 projectId: 'p1',  
 employee: [  
 {fullName: "John Doe", employeeId: 'e1'},  
 {fullName: "John Smith", employeeId: 'e2'}  
 ]  
 },  
  
 {  
 projectName: "Project2",  
 projectId: 'p2',  
 employees: [{fullName: "John Doe", employeeId: 'e1'},  
 {fullName: "Mary John", employeeId: 'e3'},  
 {fullName: "Bill Evan", employeeId: 'e4'}  
 ]  
 }  
];  
  
module.exports.lookupByProjectId = **function**(id){  
 **let** r = data.filter(object => {  
 **return** object.projectId == id;  
 });  
 **if**(r.length){  
 **return** r[0];  
 }**else**{  
 **return undefined**;  
 }  
};  
  
module.exports.lookupByEmployeeId = **function**(id){  
 **let** r = data.filter(object => {  
 **for**(**let** i = 0; i < object.employees.length;i++){  
 **if** (object.employees[i].employeeId === id) {  
  
 **return true**;  
 }  
 }  
 **return false**;  
 });  
 **if**(r.length){  
 **return** (r);  
  
 }**else**{  
 **return undefined**;  
 }  
};

<p>Project id - {{**id**}}</p>  
<p>Project Name - {{**projectName**}}</p>  
<hr>  
<p>Employees in this project:</p>  
<ol>  
 {{#**each results**}}  
 <li>  
 {{**this**.**fullName**}} ({{**this**.**employeeId**}})  
 </li>  
 {{/**each**}}  
</ol>

**const** express = require('express');  
**const** app = express();  
**var** exphbs = require('express-handlebars');  
  
//set up handlebars wiew engine  
//const handlebars = require('express-handlebars').create({defaultLayout:'main'});  
app.engine('handlebars',exphbs());  
app.set('view engine','handlebars');  
  
// app.engine('handlebars', exphbs());  
// app.set('view engine', 'handlebars');  
  
//use the project module  
**const** pm = require('./projectModule.js');  
  
//writr the code for the following route  
app.get('/project/:id', (req,res) => {  
 **let** result;  
 **let** id = (req.params.id);  
 result = (pm.lookupByProjectId(id));  
 res.format({  
 'text/html': () => {  
 res.type('text/html');  
 // console.log(JSON.stringify(result, null, 4));  
 res.render('projectView', {id:id,projectName:result.projectName,results:result.employees});  
 // "+ JSON.stringify(result, null, 4));  
 // res.send(id + result.projectName );  
 },  
 'default': () => {  
 res.status(404);  
 res.send("<b>404 - Not Found</b>");  
 }  
 });  
  
  
 // res.send('projectView',id:id,projectName:result.projectName,name:result.fullName,employeeId:result.employeeId});  
  
});  
  
app.listen(3000,()=>{  
 console.log('http://localhost:3000');  
});

Question 3:

<?php  
 $chars = array("a","b","i","c","e","f",0,"z",2);  
 $vowels = "";  
 $consonants = "";  
 $number = "";  
  
 foreach ($chars as $char){  
 if(is\_int($char) ){  
 echo $number = $number.$char;  
  
 }else if($char == "a" || $char == "i" || $char == "e") {  
 echo $vowels = $vowels.$char;  
 }  
 else{  
 echo $consonants = $consonants.$char;  
 }  
 }  
  
 echo "<br>";  
 echo "Vowels: ".$vowels."<br>";  
 echo "Consonants: ".$consonants."<br/>";  
 echo "Numbers: ".$number;

Question 4:

Rectangle class

<?php  
  
class Rectangle {  
  
 //property declarations  
 private $width;  
 private $height;  
  
 //constructor  
 public function \_\_construct($width,$height){  
 $this->width = $width;  
 $this->height = $height;  
 }  
  
 public function setWidth($width){  
 $this->width = $width;  
 }  
  
 public function getWidth(){  
 return $this->width;  
 }  
  
 public function setHeight($height){  
 $this->height = $height;  
 }  
  
 public function getHeight(){  
 return $this->height;  
 }  
  
 public function getArea() {  
 return $this->width \* $this->height;  
 }  
}  
?>

Square class

<?php  
  
class Square extends Rectangle {  
 public function \_\_construct($side) {  
 parent::*\_\_construct*($side, $side);  
 }  
  
}

<?php  
//require\_once('../database.php');  
require\_once('Rectangle.php');  
require\_once ('Square.php');  
  
 $rectangle = new Rectangle(10,20);  
  
  
 echo 'The area of width: '.$rectangle->getWidth().' and height: '.  
 $rectangle->getHeight().' is: '.$rectangle->getArea().'<br/>';  
  
  
 $square = new Square(20);  
  
 echo 'The area of width: '.$square->getWidth().' and height: '.  
 $square->getHeight().' is: '.$square->getArea();

Question 5:

<?php  
if(isset($\_POST['submit'])) {  
 //set the json file  
 $file = 'bu\_students.json';  
 //Read the json file content  
 $str\_data = file\_get\_contents($file);  
 //convert json object to php associative array  
 $json\_data = json\_decode($str\_data, true);  
  
 //remove submit from our POST array, we dont want to add it to the file  
 unset($\_POST['submit']);  
  
 //Write your code only for the missing functionality  
  
 $firstName = filter\_input(*INPUT\_POST*,'firstName');  
 $lastName = filter\_input(*INPUT\_POST*,'lastName');  
 //Validate inputs  
 if($firstName == FALSE ||$firstName == null || $lastName == FALSE || $lastName == null){  
 $error\_message = "Invalid student data. Check all fields and try again.";  
 include 'error.php';  
 }else{  
 //get form data  
 $newstudents['firstName'] = $firstName;  
 $newstudents['lastName'] = $lastName;  
 array\_push($json\_data['students'],$newstudents);  
  
 }  
  
 //Create a new student, populate it and append it to the bu\_students  
  
 $fh = fopen($file, 'w') or die("Error opening output file");  
 fwrite($fh, json\_encode($json\_data, *JSON\_UNESCAPED\_UNICODE* | *JSON\_PRETTY\_PRINT*));  
 fclose($fh);  
 header('Location:success.php');  
  
}else{  
 echo 'Sorry, nothing was submitted.';  
}  
  
?>