LAB2:

**HTML**

<h1>A Blue Heading</h1>  
  
<p>A red paragraph.</p>

**With Styles**

<h1 style="color:blue;">A Blue Heading</h1>  
  
<p style="color:red;">A red paragraph.</p>

**Links**

<a href="*url*">*link text*</a>

**Images**

<img src="link or path" alt="Flowers in Chania">

**Class**

<div class="city">  
  <h2>Kathmandu</h2>  
  <p>KTM is the capital of Nepal.</p>  
</div>

<style>  
.city {  
  background-color: tomato;  
  color: white;  
  border: 2px solid black;  
  margin: 20px;  
  padding: 20px;  
}  
</style>

**ID**

<h1 id="myHeader">My Header</h1>

<style>  
#myHeader {  
  background-color: lightblue;  
  color: black;  
  padding: 40px;  
  text-align: center;}  
</style>

<h1 class=’myheader’>My header </h1>

Css

<style>  
.myHeader {  
  background-color: lightblue;  
  color: black;  
  padding: 40px;  
  text-align: center;}  
</style>

h1 {

background-color: lightblue;  
  color: black;  
  padding: 40px;  
  text-align: center;}

}

<h1 class=’myheader’>My header

<h2>ajshd</h2>

</h1>

<style>  
.myHeader {  
  background-color: lightblue;  
  color: black;  
  padding: 40px;  
  text-align: center;}  
</style>

Myheader h2 {

background-color: lightblue;  
  color: black;  
  padding: 40px;  
  text-align: center;}

}

**CSS**

Css Selector

.center {  
  text-align: center;  
  color: red;  
}

#para1 {  
  text-align: center;  
  color: red;  
}

P .center {  
  text-align: center;  
  color: red;  
}

\* {  
  text-align: center;  
  color: blue;  
}

h1, h2, p {  
  text-align: center;  
  color: red;  
}

**JavaScript**

**Let vs const vs var**

let x="A";

* undefined

let x=0

* undefined

var x="A";

* VM267:1 Uncaught SyntaxError: Identifier 'x' has already been declared at <anonymous>:1:1

In case of var, only can be used once

In case of {}

{let y = "a";}

* undefined

console.log(y)

* Uncaught ReferenceError: y is not defined at <anonymous>:1:13

{let y = "a"; console.log(y)}

* A

var y=10;

* undefined

{var y=11;}

* undefined

console.log(y)

var x = 2;    // Allowed  
let x = 3;    // Not allowed  
  
{  
let x = 2;    // Allowed  
let x = 3     // Not allowed  
}  
  
{  
let x = 2;    // Allowed  
var x = 3     // Not allowed  
}

Allowed

carName = "Volvo";  
var carName;

Not Allowed

carName = "Saab";  
let carName = "Volvo";

const x = 10;  
// Here x is 10  
  
{  
const x = 2;  
// Here x is 2  
}  
  
// Here x is 10

Operators

let x = 100 + 50;

Datatypes

let length = 16;                               // Number  
let lastName = "Johnson";                      // String  
let x = {firstName:"John", lastName:"Doe"};    // Object

Function

function mux(p1, p2) {  
  return p1 \* p2;   // The function returns the product of p1 and p2  
}

mux(1,2);

Objects

const person = {  
  firstName: "Sahaj",  
  lastName: "Shakya",  
  age: 25

};

person.lastName;

person["lastName"];

Concat

const arr1 = ["BCT", "A"];  
const arr2 = ["BCT", "B", "2075"];  
const arr3 = arr1.concat(arr2);

filter

const BCT = [1, 2, 40, 50, 3];

function checkBCTB(no) {  
  return no >= 18;

}

const BCTB = BCT.filter(checkBCTB);

map

const numbers = [65, 44, 12, 4];

function mul(num) {  
  return num \* 2;  
}

twice = numbers.map(mul);

Slice

const worldcup = ["JAPAN", "ARGENTINA", "BRAZIL", "KOREA"];

const eliminated = worldcup.slice(1, 3);

SpreadOperator

const numbersOne = [1, 2, 3];

const numbersTwo = [4, 5, 6];

const numbersCombined = [...numbersOne, ...numbersTwo];

const numbers = [1, 2, 3, 4, 5, 6];

const [one, two, ...rest] = numbers;