```
// var obj = { name: "Piyush" };
// function sayHello(age) {
//   return "Hello " + this.name + " " + age;
// }

// console.log(sayHello());
// console.log(sayHello.call(obj, 24));

// function sayHellos(day, status) {
//   return "Hello " + this.name + " today is " + day + " and feel " + status;
// }
// sayHellos.apply(obj, ["tuesday", "good"]);

// const helloFn = sayHello.bind(obj);
// console.log(helloFn());

// //O/P question
// const person = { name: "Piyush" };

// function sayHi(age) {
//   return `${this.name} is ${age} years`;
// }

// console.log(sayHi.call(person, 24));
// console.log(sayHi.bind(person, 24));

// const age = 10;
// var person2 = {
//   name: "Aman",
//   age: 87,
//   getAge: function () {
//     return this.age;
//   },
// };
// var person3 = { age: 25 };
// person2.getAge.call(person3);

// //O/P based
// var status = "😎";
// setTimeout(() => {
//   const status = "😍";
//   const data = {
//     status: "🥹",
//     getStatus() {
//       return this.status;
//     },
//   };
//   console.log(data.getStatus());
```

```
//   console.log(data.getStatus.call(this));
// }, 0);

// const animals = [
//   { species: "Lion", name: "King" },
//   { species: "Whale", name: "Queen" },
// ];

// function printAnimals(i) {
//   this.print = function () {
//     console.log("#" + i + " " + this.species + ": " + this.name);
//   };
//   this.print();
// }
// // printAnimals.call(animals); // #undefined undefined: undefined
// for (let i = 0; i < animals.length; i++) {
//   printAnimals.call(animals[i], i);
// }

// //O/P based
// const array = ["a", "b"];
// const elements = [0, 1, 2];
// array.push(elements);
// console.log(array);
// array.push.apply(array, elements); //[a,b,0,1,2]
// console.log(array);

// const numbers = [5, 6, 2, 3, 7];
// let max = Math.max.apply(null, numbers); // equal to Math.max
// let min = Math.min.apply(null, numbers); // equal to Math.min
// (max = -Infinity), (min = +Infinity);

// for (let i = 0; i < numbers.length; i++) {
//   if (numbers[i] > max) {
//     max = numbers[i];
//   }
//   if (numbers[i] < min) {
//     min = numbers[i];
//   }
// }

// function f() {
//   console.log(this); // ?
// }

// let user = {
//   g: f.bind(null),
// };
```

```
// user.g();
// function f1() {
//   console.log(this.name);
// }
// f = f.bind({ name: "Pollo" }).bind({ name: "Kobe" });
// f();
// //Password Q

// //Arrow functions works as it is no call,apply,bind
// const ages = 10;

// var persona = {
//   name: "Hello",
//   age: 25,
//   getAgeArrow: () => console.log(this.age),
//   getAge: function () {
//     console.log(this.age);
//   },
// };
// var persona2 = { age: 76 };
// persona.getAge.call(persona2);
// persona.getAgeArrow.call(persona2);

//POLYFILL for call,bind,apply
let car1 = { color: "Red", company: "Ferrari" };
function purchaseCar(currency, price) {
  console.log(`I have purchased ${this.color} ${this.company} for ${currency}
${price}`);
}
// purchaseCar.call(car1, "Rs", "50L");
Function.prototype.myApply = function (context = {}, args = []) {
  if (typeof this !== "function") {
    throw new Error(this + "It is not callable");
  }
  if (!Array.isArray(args)) {
    throw new TypeError("CreateListFromArrayType called on Non object");
  }
  context.fn = this;
  context.fn(...args);
};
Function.prototype.myBind = function (context = {}, ...args) {
  if (typeof this !== "function") {
    throw new Error(this + "It is not callable");
  }
  context.fn = this;
  return function (...newArgs) {
    return context.fn(...newArgs);
```

```
  };
};
Function.prototype.myCall = function (context = {}, ...args) {
  if (typeof this !== "function") {
    throw new Error(this + "It is not callable");
  }
  context.fn = this;
  context.fn(...args);
};

purchaseCar.myCall(car1, "Rs", "50L");
```