

```
function getRandomInt(max) {
  return Math.floor(Math.random() * max);
}
```

```
function resolveAfter2Seconds(x) {
  return new Promise(resolve => {
    setTimeout(() => {
      resolve(x);
    },x);
  });
}
```

```
async function f3() {
  if(document.getElementById("actionButton").innerHTML=="REALIZANDO SIMULACIÓN"){
    alert("LA SIMULACIÓN YA SE ESTA EJECUTANDO");
    return;
  }

  if(document.getElementById('itered').value==0){
    alert("SELECCIONE UN NUMERO DE ITERACIONES PARA PODER COMENZAR LA SIMULACIÓN");
    return;
  }
  document.getElementById("actionButton").innerHTML="REALIZANDO SIMULACIÓN";
  var cara=0,cruz=0;
  var totIter=0;
  var prov1=0;
  var prov2=0;
  for (var i = 0; i < document.getElementById('itered').value; i++) {
    totIter++;
    var speed =document.getElementById('velocidad').value;
    console.log(`velocidad: ${speed}`);
    if(speed==""){
      await resolveAfter2Seconds(100);
      document.getElementById("iteraciones").innerHTML=`Iteraciones: ${i+1}`;
    }else if(speed==1){
      await resolveAfter2Seconds(400);
      document.getElementById("iteraciones").innerHTML=`Iteraciones: ${i+1}`;
    }else if(speed==2){
      await resolveAfter2Seconds(10);
      document.getElementById("iteraciones").innerHTML=`Iteraciones: ${i+1}`;
    }
  }
}
```

```

var num=getRandomInt(2)+1;

    if(num==1){
        cara++;
        prov1= Math.floor(cara/(i+1)*100);
        prov2=100-prov1;
        if(speed!=3){
            document.getElementById("caraProv").innerHTML=`Prov:
${prov1}%`;

            document.getElementById("caraRep").innerHTML=`Rep: ${cara}`;
            document.getElementById("cruzProv").innerHTML=`Prov:
${prov2}%`;

            document.getElementById("cruzRep").innerHTML=`Rep: ${cruz}`;
            drwarChart3(cara,cruz);
        }
    }else {
        cruz++;
        prov2=Math.floor(cruz/(i+1)*100);
        prov1=100-prov2;
        if(speed!=3){
            document.getElementById("caraProv").innerHTML=`Prov:
${prov1}%`;

            document.getElementById("caraRep").innerHTML=`Rep: ${cara}`;
            document.getElementById("cruzProv").innerHTML=`Prov:
${prov2}%`;

            document.getElementById("cruzRep").innerHTML=`Rep: ${cruz}`;
            drwarChart3(cara,cruz);
        }
    }

}

var prov1=Math.floor(cara/(totIter+1)*100);
var prov2=100-prov1;
document.getElementById("caraProv").innerHTML=`Prov: ${prov1}%`;
document.getElementById("caraRep").innerHTML=`Rep: ${cara}`;
document.getElementById("cruzProv").innerHTML=`Prov: ${prov2}%`;
document.getElementById("cruzRep").innerHTML=`Rep: ${cruz}`;

document.getElementById("iteraciones").innerHTML=`Iteraciones totales: ${totIter}`;
drwarChart3(cara,cruz);

```

```

        document.getElementById("actionButton").innerHTML="COMENZAR SIMULACIÓN";
    // 10
}

function drwarChart3(r1,r2){
    var canvas=document.getElementById('myChart').getContext('2d');
    const labels = [
        'CARA',
        'CRUZ',
    ];

    const data = {
        labels: labels,
        datasets: [{
            label:"Resutados lanzamientos de moneda",
            labels: ["Veces que salió cara","Veces que salió cruz"],
            data: [0,r1, r2],
            backgroundColor: [
                'rgba(255, 205, 86, 0.2)',
                'rgba(75, 192, 192, 0.2)',
            ],
            borderColor: [
                'rgb(255, 205, 86)',
                'rgb(75, 192, 192)',
            ],
            borderWidth: 1
        }]
    };

    const config = {
        type: 'bar',
        data: data,
        options: {
            scales: {
                y: {
                    beginAtZero: true
                }
            }
        },
    };

    try{

```

```
        var chartExist = Chart.getChart("myChart"); // <canvas> id
        if (chartExist !== undefined){
            chartExist.destroy();
        }
    }catch(e){
        console.log(e);
    }
}
```

```
const myChart = new Chart(
    document.getElementById('myChart'),
    config
);
}
```