

Presupuesto Disponible

\$65,000.00

INGRESOS +100,000.00

EGRESOS -\$35,000.00

INGRESOS	
Honorarios	+100,000.00

EGRESOS	
Tarjeta Crédito	-\$20,000.00 30.9%
Alquiler	-\$15,000.00 46.9%

Alquiler 15000

Javascript

```
const ingresos = [
  new Ingreso('Salario', 1000),
  new Ingreso('Venta coche', 200),
  new Ingreso('Honorarios', 500)
];

const egresos = [
  new Egreso('Expensas', 30000),
  new Egreso('Comida Perros', 17000),
  new Egreso('Arba', 1500),
  new Egreso('Municipalidad', 7500)
];

let cargarApp = () =>{
  cargarCabecero();
  totalIngresos();
  totalEgresos();
  cargarEgresos();
}

function totalIngresos() {
  let totalIngreso = 0;
  for (let ingreso of ingresos) {
    totalIngreso += ingreso.valor;
  }
  return totalIngreso;
}

let totalEgresos = () => {
  let totalEgreso = 0;
  for (let egreso of egresos){
    totalEgreso += egreso.valor;
  }
  return totalEgreso;
}

let cargarCabecero = () =>{
  let presupuesto = totalIngresos() - totalEgresos();
  let porcentajeEgreso = totalEgresos()/totalIngresos();
  document.getElementById('presupuesto').innerHTML =
    formatoMoneda(presupuesto);
  document.getElementById('porcentaje').innerHTML =
    formatoPorcentaje(porcentajeEgreso);
  document.getElementById('ingresos').innerHTML = '+' +
    formatoMoneda(totalIngresos());
}
```

```

        document.getElementById('egresos').innerHTML='.' +
formatoMoneda(totalEgresos());
    }

    const formatoMoneda = (valor) =>{
        return valor.toLocaleString('en-US',{style:'currency',
currency:'USD', minimumFractionDigits:2});
    }

    const formatoPorcentaje = (valor) =>{
        return valor.toLocaleString('en-US',{style:'percent',
minimumFractionDigits:1});
    }

    const cargarIngresos = ()=>{
        let ingresosHTML = "";
        for(let ingreso of ingresos){
            ingresosHTML += crearIngresoHTML(ingreso);
        }
        document.getElementById("lista-ingresos").innerHTML =
ingresosHTML;
    }

    const crearIngresoHTML = (ingreso) =>{
        let ingresoHTML = `
        <div class="elemento limpiarEstilos">
        <div
class="elemento_descripcion">${ingreso.descripcion}</div>
        <div class="derecha limpiarEstilos">
        <div
class="elemento_valor">+${formatoMoneda(ingreso.valor)}<
/div>
        <div class="elemento_eliminar">
        <button class="elemento_eliminar--btn">
        <ion-icon name="close-circle-outline"
onclick="eliminarIngreso(${ingreso.id})"></ion-
icon>
        </button>
    `
    }

```

```

    </div>
    </div>
    </div>
    `;
    return ingresoHTML;
}

    const eliminarIngreso=(id)=>{
        let indiceEliminar = ingresos.findIndex(ingreso =>
ingreso.id ===id);

        ingresos.splice(indiceEliminar, 1);
        cargarCabecero();
        cargarIngresos();
    }

    const cargarEgresos = ()=>{
        let egresosHTML = "";
        for(let egreso of egresos){
            egresosHTML += crearEgresoHTML(egreso);
        }
        document.getElementById("lista-egresos").innerHTML
= egresosHTML;
    }

    const crearEgresoHTML = (egreso) =>{
        let egresoHTML = `
        <div class="elemento limpiarEstilos">
        <div
class="elemento_descripcion">${egreso.descripcion}</div>
        <div class="derecha limpiarEstilos">
        <div class="elemento_valor">-
${formatoMoneda(egreso.valor)}</div>
        <div
class="elemento_porcentaje">${formatoPorcentaje(egreso.v
alor/totalEgresos())}</div>
        <div class="elemento_eliminar">
        <button class="elemento_eliminar--btn">
        <ion-icon name="close-circle-outline"
    `
    }

```

```

        onclick="eliminarEgreso(${egreso.id})"></ion-
icon>
        </button>
        </div>
        </div>
        </div>
        `;
        return egresoHTML;
    }

    const eliminarEgreso = (id) =>{
        let indiceEliminar = egresos.findIndex(egreso =>
egreso.id === id);

        egresos.splice(indiceEliminar, 1);
        cargarCabecero();
        cargarEgresos();
    }

    let agregarDato()=>{
        let forma = document.forms['forma'];
        let tipo = forma['tipo'];
        let descripcion = forma['descripcion'];
        let valor = forma['valor'];
        if(descripcion.value !== "" && valor.value !=="){
            if(tipo.value === 'ingreso'){
                ingresos.push(new Ingreso(descripcion.value,
+valor.value))
                cargarCabecero();
                cargarIngresos();
            }
        }
        } else if(tipo.value === 'egreso'){
            egresos.push(new Egreso(descripcion.value,
+valor.value))
            cargarCabecero();
            cargarEgresos();
        }
    }
}

```