

## Module 30.5-Integrate JavaScript Bonus Content

### 30\_5-1 Revisit Baap er bank with Clean functional approach

```
function getInputValue(fieldId)
{
    const inputField=document.getElementById(fieldId);
    const valueInText=inputField.value;
    const value=parseFloat(valueInText);

    inputField.value="";
    return value;
}

function updateTotal(fieldId,amount)
{
    const totalTag=document.getElementById(fieldId);
    const previousTotalInText=totalTag.innerText;
    const previousTotal=parseFloat(previousTotalInText);
    const newTotal=previousTotal+amount;
    totalTag.innerText=newTotal;
}

document.getElementById("deposit-
button").addEventListener("click",function(){
    const amount = getInputValue("deposit-input");
    updateTotal("deposit-total",amount);
});
```

### 30\_5-2 Share function for deposit and withdraw and balance

```
function getInputValue(fieldId) {
    const inputField = document.getElementById(fieldId);
    const valueInText = inputField.value;
    const value = parseFloat(valueInText);

    inputField.value = "";
    return value;
}

function updateTotal(fieldId, amount) {
    const totalTag = document.getElementById(fieldId);
```

```

    const previousTotalInText = totalTag.innerText;
    const previousTotal = parseFloat(previousTotalInText);
    const newTotal = previousTotal + amount;
    totalTag.innerText = newTotal;
}

function updateBalance(amount, isAdding) {
    const balanceTag = document.getElementById("balance-total");
    const balanceInTag = balanceTag.innerText;
    const previousBalance = parseFloat(balanceInTag);
    let newBalance;

    if(isAdding==true)
    {
        newBalance = previousBalance + amount;
    }
    else
    {
        newBalance = previousBalance - amount;
    }
    balanceTag.innerText = newBalance;
}

document
    .getElementById("deposit-button")
    .addEventListener("click", function () {
        const amount = getInputValue("deposit-input");
        updateTotal("deposit-total", amount);
        updateBalance(amount, true);
    });

document
    .getElementById("withdraw-button")
    .addEventListener("click", function () {
        const amount = getInputValue("withdraw-input");
        updateTotal("withdraw-total", amount);
        updateBalance(amount, false);
    });

```

### 30\_5-3 Error handling, more functions, different balance calculation

```

function getInputValue(fieldId) {
    const inputField = document.getElementById(fieldId);
    const valueInText = inputField.value;

```

```

    const value = parseFloat(valueInText);

    inputField.value = "";
    return value;
}

function getInnerTextValue(fieldId) {
    const fieldTag = document.getElementById(fieldId);
    const fieldValueInText = fieldTag.innerText;
    const value = parseFloat(fieldValueInText);
    return value;
}

function updateTotal(fieldId, amount) {
    const previousTotal = getInnerTextValue(fieldId);
    const newTotal = previousTotal + amount;
    document.getElementById(fieldId).innerText = newTotal;
}

function updateBalance(amount, isAdding) {
    const previousBalance = getInnerTextValue("balance-total");
    let newBalance;

    if (isAdding == true) {
        newBalance = previousBalance + amount;
    } else {
        newBalance = previousBalance - amount;
    }
    document.getElementById("balance-total").innerText = newBalance;
}

document
    .getElementById("deposit-button")
    .addEventListener("click", function () {
        const amount = getInputValue("deposit-input");

        if (amount > 0) {
            updateTotal("deposit-total", amount);
            updateBalance(amount, true);
        }
    });

document
    .getElementById("withdraw-button")
    .addEventListener("click", function () {

```

```

const amount = getInputValue("withdraw-input");
const balance = getInnerTextValue("balance-total");

if (amount > 0 && amount <= balance) {
    updateTotal("withdraw-total", amount);
    updateBalance(amount, false);
} else
{
    alert("You can not withdraw more than what you have in your
account");
}
});

```

### 30\_5-4 Revisit id, class selector, querySelector, querySelectorAll

```

const players = document.getElementsByClassName("player");
for(const player of players)
{
    player.style.border='5px solid red';
    player.style.margin='10px';
    player.style.padding="10px";
    player.style.borderRadius="15px";
}

function addPlayer(){
    const playeContainer=document.getElementById('playes');
}

```

### 30\_5-5 (advanced) Event Bubble in a semi complex situation

```

function setPlayerStyle(player) {
    player.style.border = "5px solid red";
    player.style.margin = "10px";
    player.style.padding = "10px";
    player.style.borderRadius = "15px";
}

const players = document.getElementsByClassName("player");
for (const player of players) {
    setPlayerStyle(player);
    player.addEventListener("click", function () {

```

```

        player.style.backgroundColor = "yellow";
    });
}

function addPlayer() {
    const playesContainer = document.getElementById("players");
    const players = document.createElement("div");
    players.classList.add("player");
    players.innerHTML = `
        <h4 class="player-name">New Player</h4>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Nobis,
alias repellat fuga neque iure ipsa quaerat et ut praesentium
repudiandae, ullam omnis debitis quasi vero deserunt aliquam, ab
perspiciatis est!</p>
    `;
    setPlayerStyle(players);
    playesContainer.appendChild(players);
}

document.getElementById("players").addEventListener("click", function
(event) {
    if (event.target.tagName.toLowerCase() == "div") {
        event.target.style.backgroundColor = "yellow";
    } else {
        event.target.parentNode.style.backgroundColor = "yellow";
    }
});

```