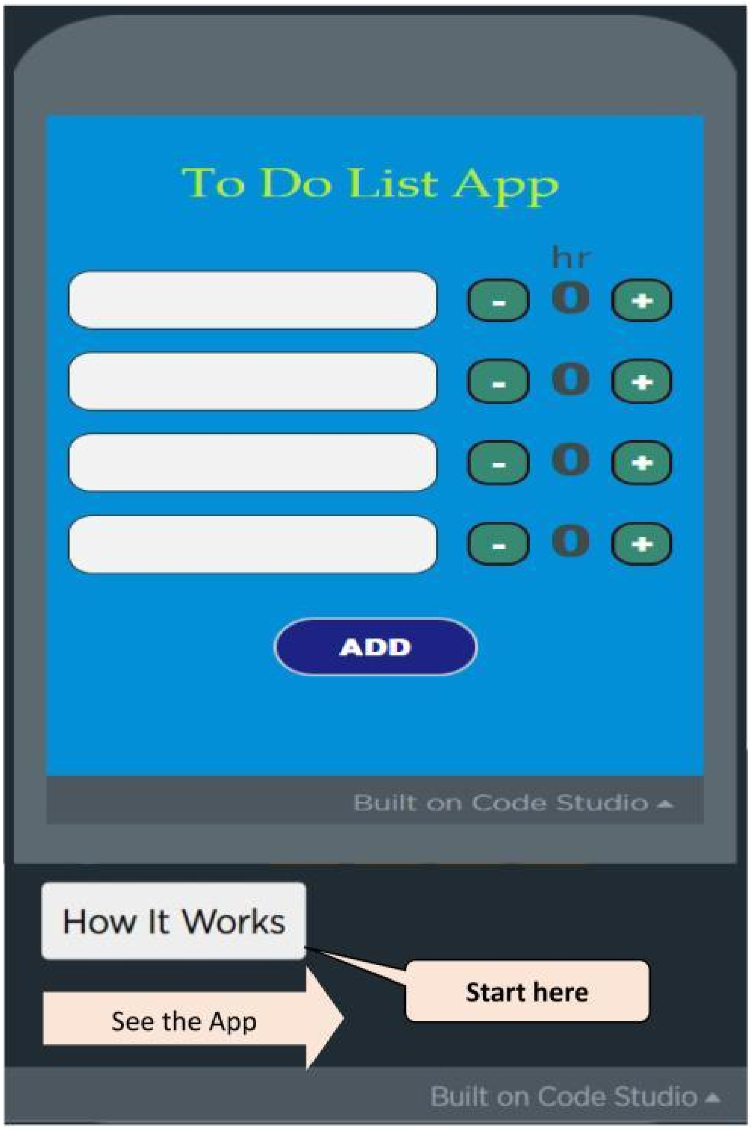
Project 63

DP EDUCATION

Coding School



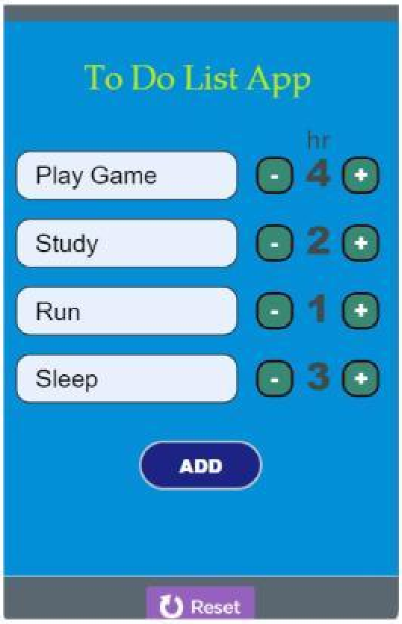
Let's create a To Do List App.

here,

The required tasks can be entered by text input on Screen1.

The time for each task can be given in hours as required by the + button and the – button.

When you click the Add button, you should go to Screen2.



To Do List App

Play Game 4

Study 2

Run 1

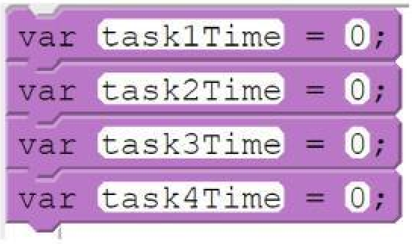
Sleep 3

(Screen1)

❖ All design related to Screen1 is given.

❖ When + button is clicked, the number of hours should increase by one in relation to each task. For that, let's first take the number of hours as 0 for each task. Let's create 4 separate variables for each task.

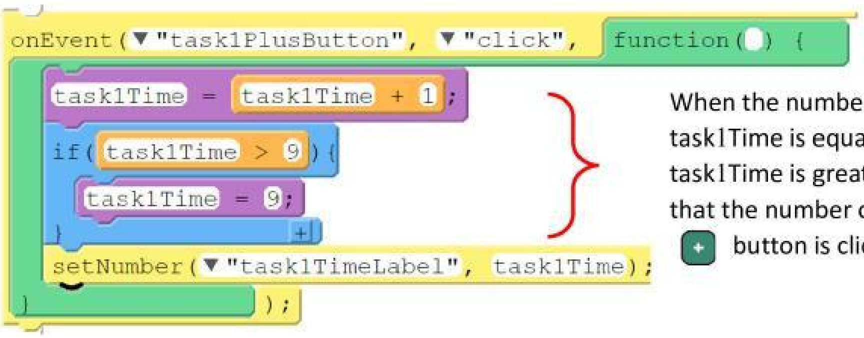
Let's create 4 variables
as "task1Time" for the first task,
"task2Time" for the second task,
"task3Time" for the third task, and
"task4Time" for the fourth task.



```
var task1Time = 0;
var task2Time = 0;
var task3Time = 0;
var task4Time = 0;
```

For the first task,

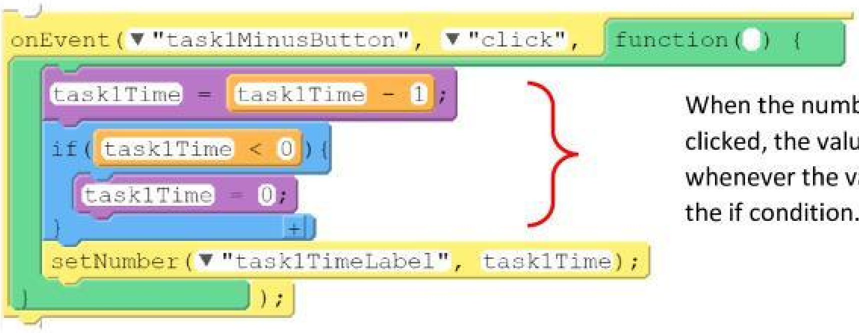
❖ When + Button ("task1PlusButton") is clicked, the hours of the first task should increase by one. Accordingly, every time + Button is clicked, the "task1Time" variable should increase one by one. And the value of "task1Time" variable should be set to "task1TimeLabel" label.



```
onEvent(▼"task1PlusButton", ▼"click", function( ) {
  task1Time = task1Time + 1;
  if(task1Time > 9){
    task1Time = 9;
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

When the number of hours is 9, the value of task1Time is equal to 9 whenever the value of task1Time is greater than 9 by the if condition so that the number of hours does not increase when + button is clicked.

- When - button is clicked ("task1MinusButton"), the number of hours in the first task should be reduced by one. Accordingly, - "task1Time" variable should decrease by one every time the button is clicked. And the value of "task1Time" variable should be set to "task1TimeLabel" label.



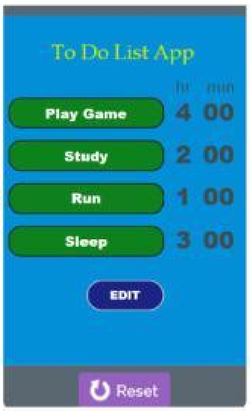
```
onEvent(▼"task1MinusButton", ▼"click", function( ) {
  task1Time = task1Time - 1;
  if(task1Time < 0){
    task1Time = 0;
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

When the number of hours is 0 - When - button is clicked, the value of task1Time is equal to 0 whenever the value of task1Time is less than 0 by the if condition.

- Do the coding for + Button and - Button for all the tasks as above.

- When the "ADD" button is clicked, it should go to screen 2 and the tasks entered in screen 1 and the time related to each task should be set to the buttons and labels in screen 2.

The tasks entered in screen1 should be displayed in the buttons.



To Do List App

hr min

Play Game 4 00

Study 2 00

Run 1 00

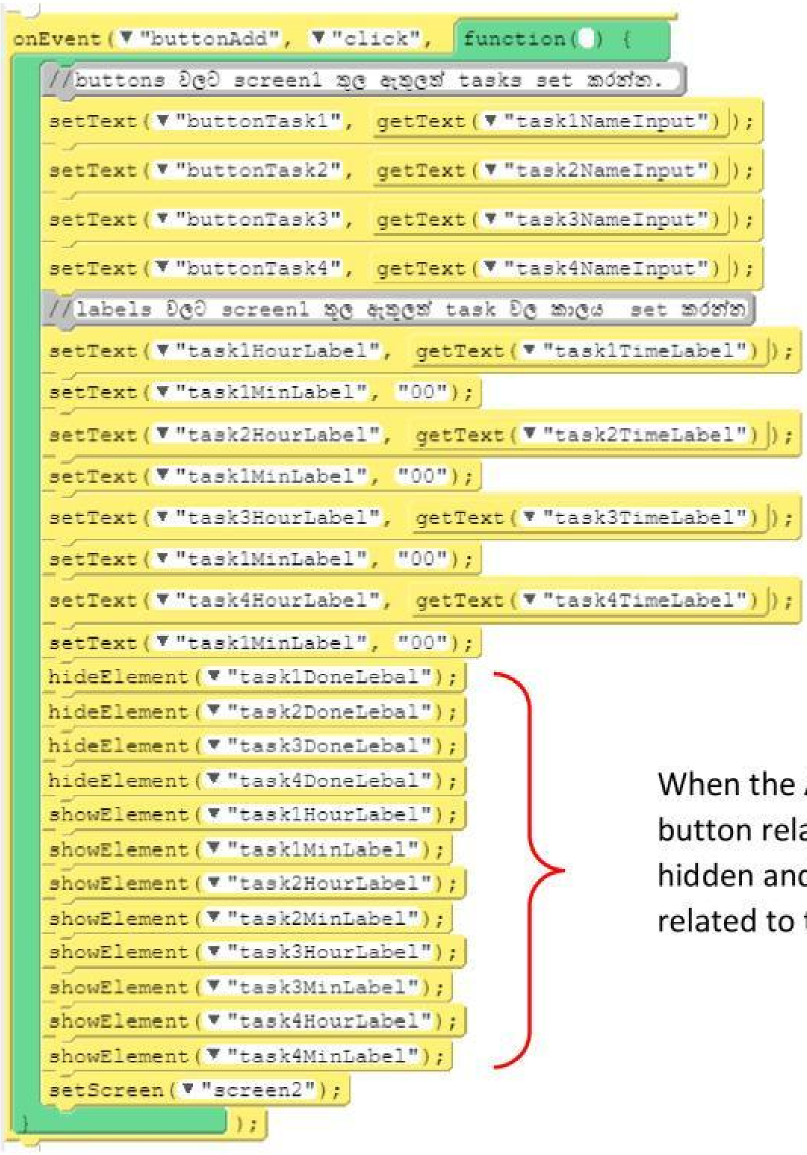
Sleep 3 00

EDIT

Reset

The time related to the task entered in screen1 should be displayed in the label.

(Screen2)

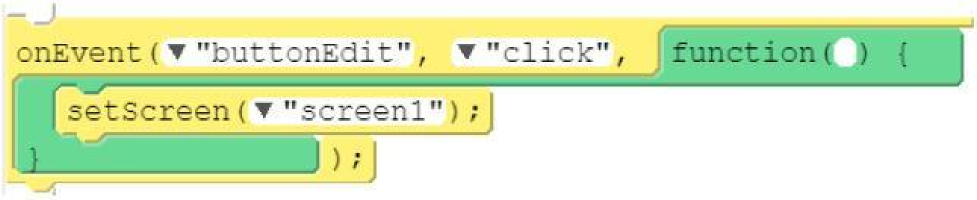


```
onEvent("buttonAdd", "click", function( ) {
  //buttons වලට screen1 තුල ඇතුලත් tasks set කරන්න.
  setText("buttonTask1", getText("task1NameInput"));
  setText("buttonTask2", getText("task2NameInput"));
  setText("buttonTask3", getText("task3NameInput"));
  setText("buttonTask4", getText("task4NameInput"));
  //labels වලට screen1 තුල ඇතුලත් task වල කාලය  set කරන්න
  setText("task1HourLabel", getText("task1TimeLabel"));
  setText("task1MinLabel", "00");
  setText("task2HourLabel", getText("task2TimeLabel"));
  setText("task1MinLabel", "00");
  setText("task3HourLabel", getText("task3TimeLabel"));
  setText("task1MinLabel", "00");
  setText("task4HourLabel", getText("task4TimeLabel"));
  setText("task1MinLabel", "00");
  hideElement("task1DoneLebal");
  hideElement("task2DoneLebal");
  hideElement("task3DoneLebal");
  hideElement("task4DoneLebal");
  showElement("task1HourLabel");
  showElement("task1MinLabel");
  showElement("task2HourLabel");
  showElement("task2MinLabel");
  showElement("task3HourLabel");
  showElement("task3MinLabel");
  showElement("task4HourLabel");
  showElement("task4MinLabel");
  setScreen("screen2");
});
```

When the Add button is clicked, the done button related to each task should be hidden and the labels showing the time related to those tasks should be shown.

❖ All the designs related to Screen2 have been given.

❖ When you click the edit button, you should go to screen1.

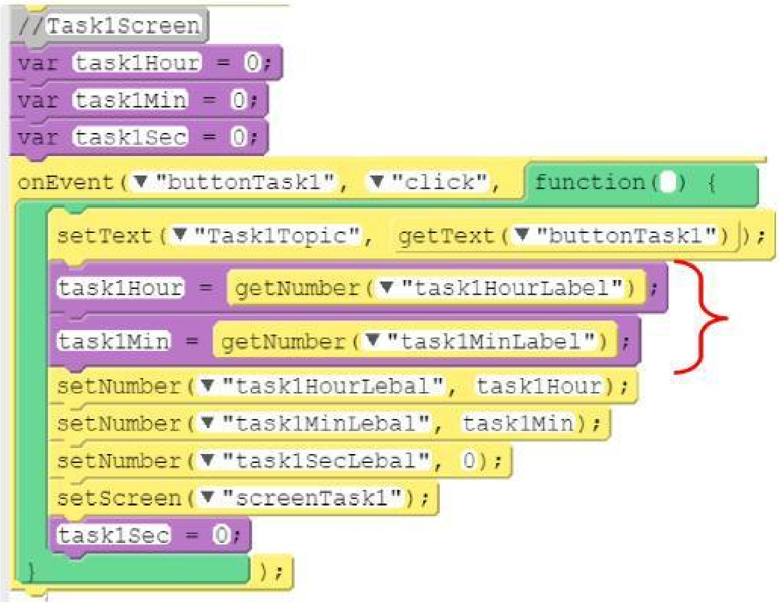


```
onEvent("buttonEdit", "click", function( ) {
  setScreen("screen1");
});
```

❖ When you click the button related to a task, you should go to the screen related to that button.

- To calculate the time related to the first task, create 3 variables as task1Hour, task1Min and task1Sec.

- When you click the "task1StartButton" button related to the first task, you should go to the "screenTask1" screen.



```
//Task1Screen
var task1Hour = 0;
var task1Min = 0;
var task1Sec = 0;
onEvent(▼"buttonTask1", ▼"click", function( ) {
  setText(▼"Task1Topic", getText(▼"buttonTask1"));
  task1Hour = getNumber(▼"task1HourLabel");
  task1Min = getNumber(▼"task1MinLabel");
  setNumber(▼"task1HourLebal", task1Hour);
  setNumber(▼"task1MinLebal", task1Min);
  setNumber(▼"task1SecLebal", 0);
  setScreen(▼"screenTask1");
  task1Sec = 0;
});
```

Assign the value of the “task1HourLabel” label to the “task1Hour” variable and the “task1MinLabel” label to the task1Min variable.

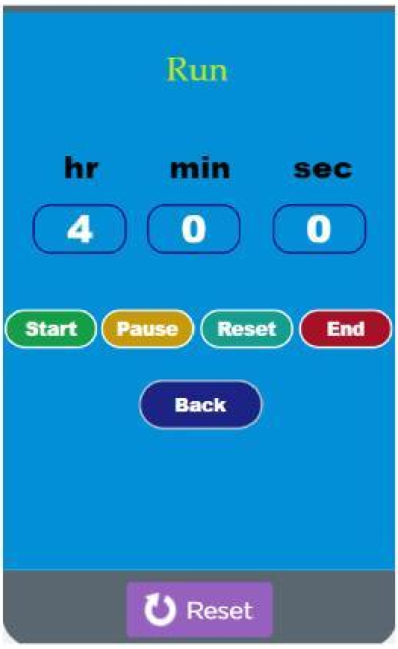
- All design related to screenTask1 is given.

When the start button is clicked, the timer should start.

The timer should stop when the Pause Button is clicked.

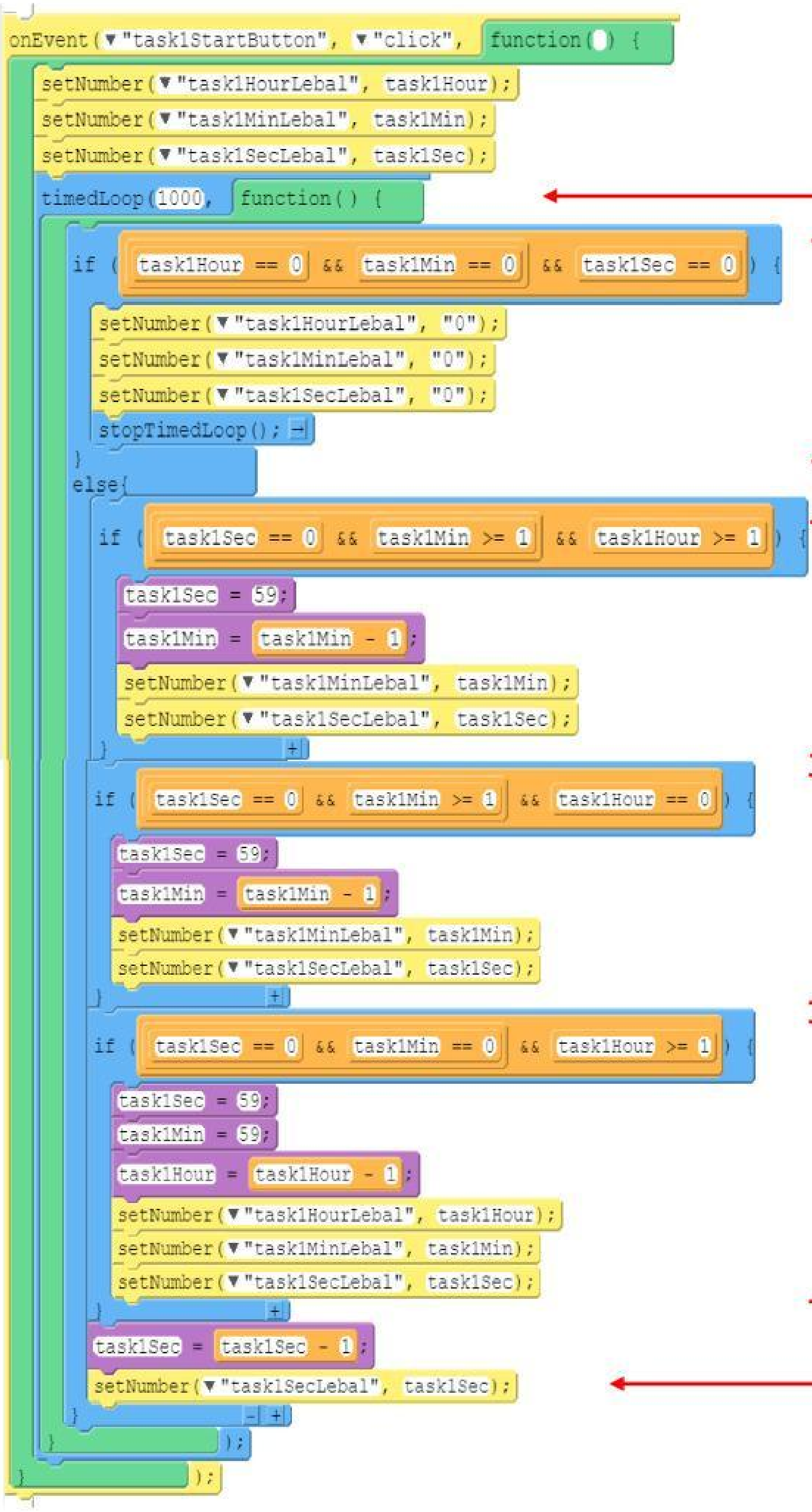
When the Reset Button is clicked, the timer should go back to the original

When the End Button is clicked, the timer should end.



(screenTask1)

- When the “task1StartButton” button is clicked, the timer should start. Use the following code block for that.



```
onEvent("task1StartButton", "click", function( ) {
  setNumber("task1HourLebal", task1Hour);
  setNumber("task1MinLebal", task1Min);
  setNumber("task1SecLebal", task1Sec);
  timedLoop(1000, function() {
    if (task1Hour == 0 && task1Min == 0 && task1Sec == 0) {
      setNumber("task1HourLebal", "0");
      setNumber("task1MinLebal", "0");
      setNumber("task1SecLebal", "0");
      stopTimedLoop();
    }
    else{
      if (task1Sec == 0 && task1Min >= 1 && task1Hour >= 1) {
        task1Sec = 59;
        task1Min = task1Min - 1;
        setNumber("task1MinLebal", task1Min);
        setNumber("task1SecLebal", task1Sec);
      }
      if (task1Sec == 0 && task1Min >= 1 && task1Hour == 0) {
        task1Sec = 59;
        task1Min = task1Min - 1;
        setNumber("task1MinLebal", task1Min);
        setNumber("task1SecLebal", task1Sec);
      }
      if (task1Sec == 0 && task1Min == 0 && task1Hour >= 1) {
        task1Sec = 59;
        task1Min = 59;
        task1Hour = task1Hour - 1;
        setNumber("task1HourLebal", task1Hour);
        setNumber("task1MinLebal", task1Min);
        setNumber("task1SecLebal", task1Sec);
      }
      task1Sec = task1Sec - 1;
      setNumber("task1SecLebal", task1Sec);
    }
  });
});
```

The code in the block is looped every second of the time loop. (1000ms = 1s)

Whenever the value of task1Hour, task1Min and task1Sec variables is 0, “task1HourLebal”, “task1MinLebal” and “task1SecLebal” labels should be displayed as 0. And the Timed loop should stop.

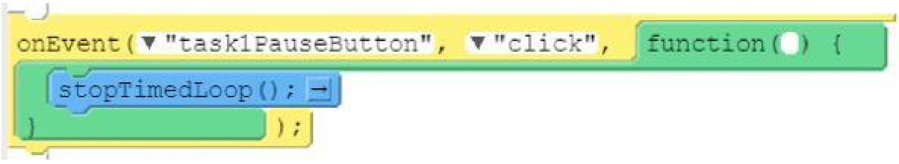
Whenever the value of the “task1Sec” variable is 0 and the value of the “task1Min” variable and the value of the “task1Hour” variable are equal to or greater than 1, the minutes should be reduced by one. Also, the number of seconds should be 59.

Whenever the value of the “task1Sec” variable and the value of the “task1Hour” variable are equal to 0 and the value of the “task1Min” variable is equal to or greater than 1, it should decrease by one minute. Also, the number of seconds should be 59.

Whenever the value of the “task1Sec” variable and the value of the “task1Min” variable are 0, the value of the “task1Hour” variable should be reduced by one hour when the value of the “task1Hour” variable is greater than or equal to 1. Also, the number of minutes and seconds should be 59.

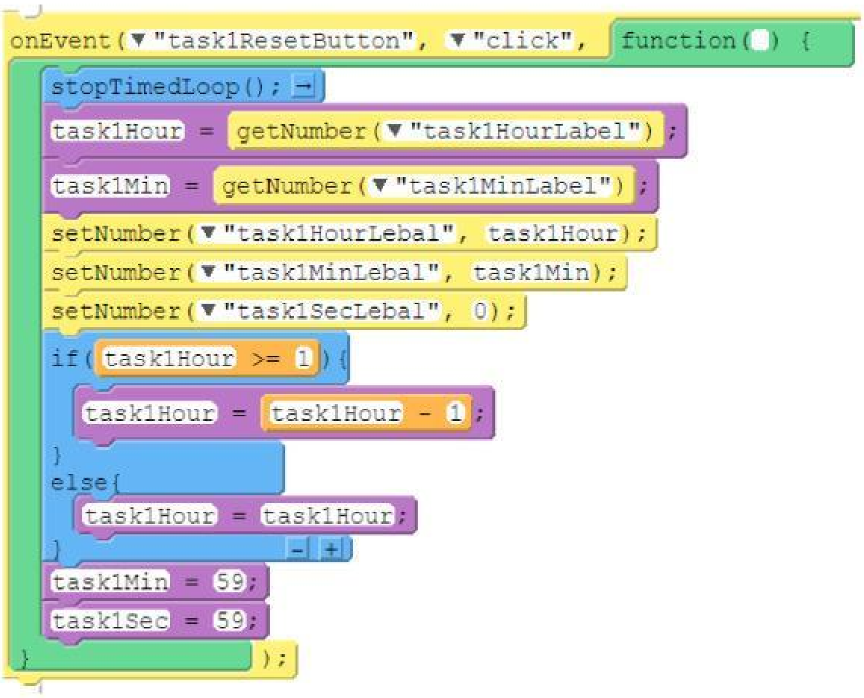
In every second, the timer should decrease by one second. Because in the timed loop, it is done by this code block.

- The timer should stop when "task1ResetButton" is clicked. Use the following code block for that.



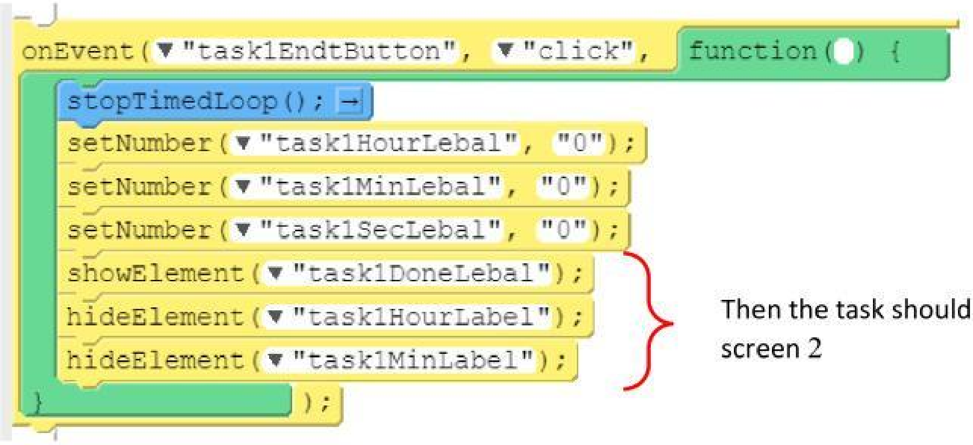
```
onEvent(▼"task1PauseButton", ▼"click", function( ) {
  stopTimedLoop();
});
```

- When "task1ResetButton" is clicked, the Timer should go back to the previously given value. Use the following code block for that.



```
onEvent(▼"task1ResetButton", ▼"click", function( ) {
  stopTimedLoop();
  task1Hour = getNumber(▼"task1HourLabel");
  task1Min = getNumber(▼"task1MinLabel");
  setNumber(▼"task1HourLebal", task1Hour);
  setNumber(▼"task1MinLebal", task1Min);
  setNumber(▼"task1SecLebal", 0);
  if(task1Hour >= 1) {
    task1Hour = task1Hour - 1;
  }
  else{
    task1Hour = task1Hour;
  }
  task1Min = 59;
  task1Sec = 59;
});
```

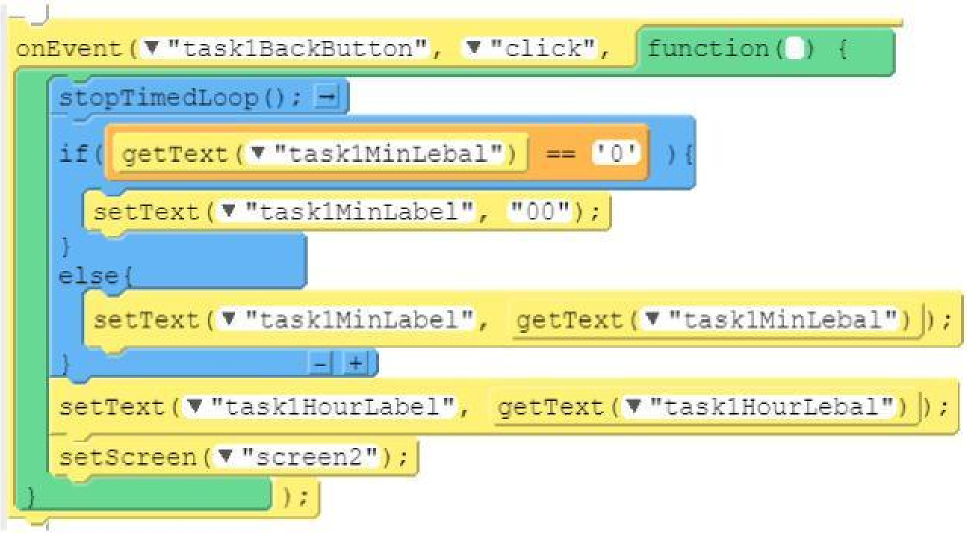
- The timer should end when the End Button is clicked.



```
onEvent(▼"task1EndtButton", ▼"click", function( ) {
  stopTimedLoop();
  setNumber(▼"task1HourLebal", "0");
  setNumber(▼"task1MinLebal", "0");
  setNumber(▼"task1SecLebal", "0");
  showElement(▼"task1DoneLebal");
  hideElement(▼"task1HourLabel");
  hideElement(▼"task1MinLabel");
});
```

Then the task should be displayed as done on screen 2

- ❖ When the back button is clicked, it should go to screen2 and its time should change according to the values in the variables.

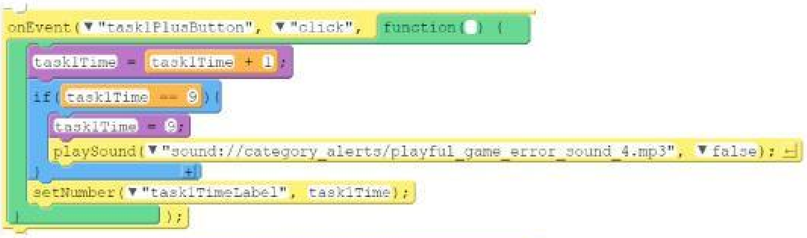


```
onEvent(▼"task1BackButton", ▼"click", function( ) {
  stopTimedLoop();
  if( getText(▼"task1MinLebal") == '0' ){
    setText(▼"task1MinLabel", "00");
  }
  else{
    setText(▼"task1MinLabel", getText(▼"task1MinLebal"));
  }
  setText(▼"task1HourLabel", getText(▼"task1HourLebal"));
  setScreen(▼"screen2");
});
```

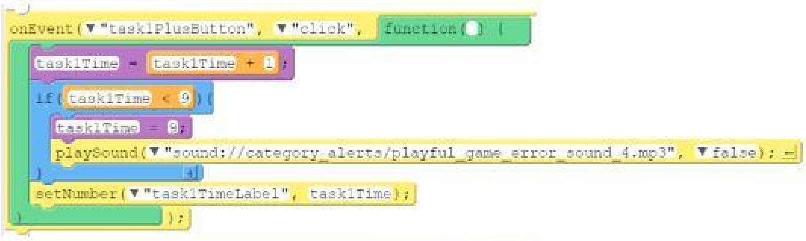
- ❖ Do the coding for each screen for all the tasks as above.

Select The correct answer

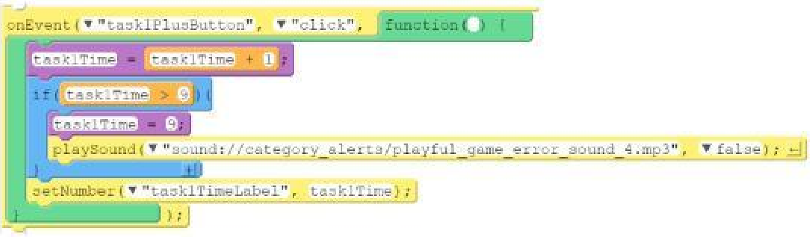
1. When the number of hours is 9, when + button is clicked, what blocks are used to prevent the number of hours from increasing and then making a sound?



```
onEvent(▼"task1PlusButton", ▼"click", function( ) {
  task1Time = task1Time + 1;
  if( task1Time == 9 ){
    task1Time = 9;
    playSound(▼"sound://category_alerts/playful_game_error_sound_4.mp3", ▼false);
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

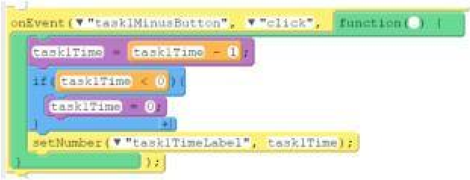


```
onEvent(▼"task1PlusButton", ▼"click", function( ) {
  task1Time = task1Time + 1;
  if( task1Time < 9 ){
    task1Time = 9;
    playSound(▼"sound://category_alerts/playful_game_error_sound_4.mp3", ▼false);
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

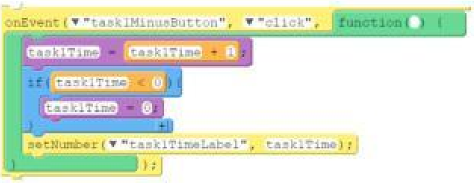


```
onEvent(▼"task1PlusButton", ▼"click", function( ) {
  task1Time = task1Time + 1;
  if( task1Time > 9 ){
    task1Time = 9;
    playSound(▼"sound://category_alerts/playful_game_error_sound_4.mp3", ▼false);
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

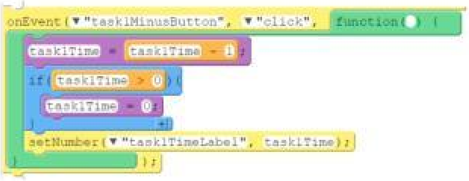
2. Select the one used so that its value is not less than 0 when clicked on the button named "tast1MinusButton"



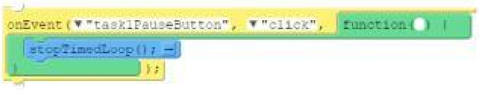
```
onEvent(▼"task1MinusButton", ▼"click", function( ) {
  task1Time = task1Time - 1;
  if( task1Time < 0 ) {
    task1Time = 0;
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```



```
onEvent(▼"task1MinusButton", ▼"click", function( ) {
  task1Time = task1Time + 1;
  if( task1Time < 0 ) {
    task1Time = 0;
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```



```
onEvent(▼"task1MinusButton", ▼"click", function( ) {
  task1Time = task1Time - 1;
  if( task1Time > 0 ) {
    task1Time = 0;
  }
  setNumber(▼"task1TimeLabel", task1Time);
});
```

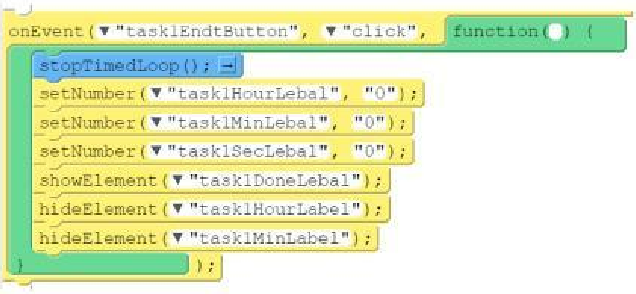
3.  What does this block do?

```
onEvent(▼"task1PauseButton", ▼"click", function( ) {
  stopTimedLoop();
});
```

When the "task1pauseButton" button is clicked, the time calculation starts.

When the "task1pauseButton" button is clicked, the time calculation will stop.

When the "task1pauseButton" button is clicked, the time calculation will stop for 5 seconds and start again.

4.  What does this block do?

```
onEvent(▼"task1EndtButton", ▼"click", function( ) {
  stopTimedLoop();
  setNumber(▼"task1HourLebal", "0");
  setNumber(▼"task1MinLebal", "0");
  setNumber(▼"task1SecLebal", "0");
  showElement(▼"task1DoneLebal");
  hideElement(▼"task1HourLabel");
  hideElement(▼"task1MinLabel");
});
```

When the "task1EndtButton" button is clicked, the time calculation starts.

When the "task1EndtButton" button is clicked, only the time is calculated.

When the "task1EndtButton" button is clicked, the time calculation stops and the "task1DoneLabale" label is shown.