var myObj = {

  gift:"pony",

  pet:"kitten",

  bed:"sleigh"};

function checkObj(myObj,checkProp) {

  // Only change code below this line

  if(myObj.hasOwnProperty(checkProp)) {

    return myObj[checkProp];

  } else {return "Not Found"};

  // Only change code above this line

}

checkObj("gift")

const myMusic = [

  {

    "artist": "Billy Joel",

    "title": "Piano Man",

    "release\_year": 1973,

    "formats": [

      "CD",

      "8T",

      "LP"

    ],

    "gold": true

  },

  {

    "artist":"Taras Shevchenko",

    "title": "Hero and symbol of Ukraine",

    "release\_year": 1814,

    "formats": [

      "Gypsy fortune teller",

      "Peasant family"

    ]

  }

];

const myStorage = {

  "car": {

    "inside": {

      "glove box": "maps",

      "passenger seat": "crumbs"

     },

    "outside": {

      "trunk": "jack"

    }

  }

};

const gloveBoxContents = myStorage.car.inside["glove box"];

const myPlants = [

  {

    type: "flowers",

    list: [

      "rose",

      "tulip",

      "dandelion"

    ]

  },

  {

    type: "trees",

    list: [

      "fir",

      "pine",

      "birch"

    ]

  }

];

const secondTree = myPlants[1].list[1];

// Setup

const recordCollection = {

  2548: {

    albumTitle: 'Slippery When Wet',

    artist: 'Bon Jovi',

    tracks: ['Let It Rock', 'You Give Love a Bad Name']

  },

  2468: {

    albumTitle: '1999',

    artist: 'Prince',

    tracks: ['1999', 'Little Red Corvette']

  },

  1245: {

    artist: 'Robert Palmer',

    tracks: []

  },

  5439: {

    albumTitle: 'ABBA Gold'

  }

};

// Only change code below this line

function updateRecords(records, id, prop, value) {

  if(prop !== 'tracks' && value !==""){

    records[id][prop] = value;

  } else if(prop==='tracks' && records[id].hasOwnProperty("tracks") === false){

    records[id][prop] = [value];

  }else if(prop==='tracks' && value!==""){

    records[id].tracks.push(value);

  }else if(value===""){

    delete records[id][prop];

  }

  return records;

}

updateRecords(recordCollection, 5439, 'artist', 'ABBA');

// Setup

const myArray = [];

let i=5;

while(i>=0) {

  myArray.push(i);

  i--;

}

// Only change code below this line

// Setup

const myArray = [];

for(var i = 1; i<=5;i++) {

myArray.push(i);

}

// Only change code below this line

// Setup

const myArray = [];

for(var i = 1; i<11;i+=2){

  myArray.push(i);

}

// Only change code below this line

// Setup

const myArray = [];

for(var i=9;i>0;i-=2){

  myArray.push(i)

}

// Only change code below this line

// Setup

const myArr = [2, 3, 4, 5, 6];

let total = 0;

for(var i =0; i<myArr.length;i++) {

  console.log(myArr[i])

  console.log(total+=myArr[i])

}

// Only change code below this line

const myArray = [];

let i = 10;

// Only change code below this line

do {

myArray.push(i);

i++

} while (i < 5)

// Setup

const contacts = [

  {

    firstName: "Akira",

    lastName: "Laine",

    number: "0543236543",

    likes: ["Pizza", "Coding", "Brownie Points"],

  },

  {

    firstName: "Harry",

    lastName: "Potter",

    number: "0994372684",

    likes: ["Hogwarts", "Magic", "Hagrid"],

  },

  {

    firstName: "Sherlock",

    lastName: "Holmes",

    number: "0487345643",

    likes: ["Intriguing Cases", "Violin"],

  },

  {

    firstName: "Kristian",

    lastName: "Vos",

    number: "unknown",

    likes: ["JavaScript", "Gaming", "Foxes"],

  },

];

function lookUpProfile(name, prop) {

  // Only change code below this line

  for(let i =0;i<contacts.length;i++) {

    if(contacts[i]['firstName']===name) {

      if(contacts[i].hasOwnProperty(prop)){

        return contacts[i][prop];

      }else{ return "No such property";

      }

      }

    }

  return "No such contact"

  // Only change code above this line

}

lookUpProfile("Akira", "likes");

function randomFraction() {

 return Math.random()\*10

}

function randomWholeNum() {

  return Math.floor(Math.random()\*10);

}

function randomRange(myMin, myMax) {

  return Math.floor(Math.random() \* (myMax - myMin + 1)) + myMin

}

function convertToInteger(str) {

return parseInt(str);

}

convertToInteger("56");

//переводить у десяткову систему

function convertToInteger(str,redix) {

return parseInt(str,2)

}

convertToInteger("10011");

function checkEqual(a, b) {

return a == b ? "Equal" : "Not Equal"

}

checkEqual(1, 2);

function checkSign(num) {

return (num > 0) ? "positive"

: (num < 0) ? "negative"

: "zero"

}

checkSign(10);

function countdown(n){

  if(n<1){

  return [];

} else { const masuv = countdown(n-1);

masuv.unshift(n)

  return masuv;

}

}

function rangeOfNumbers(startNum, endNum) {

  if(endNum - startNum == 0 ){

    return [startNum];

  }else{

    let numbers =rangeOfNumbers(startNum, endNum - 1);

    numbers.push(endNum);

    return numbers;

    }

}