Hands – On Lab

Workshop 3

AREA OF TRIANGLE

Write a function that takes the base and height of a triangle and return its area.

Example:

Areaoftriangle (3, ⟶ 4) 6

Areaoftriangle (7, ⟶ 8)

28 Notes

- Area of triangle is (base * height)/2
- Don't forget to return the result

BASKETBALL POINTS

You are counting points for a basketball game, given the amount of 2 – pointer scored and 3 – pointer scored, find the final points for the team and return the value.

Example:

points ⟶ (3,5) 3*2 + 5*3 =

21 points ⟶ (1,1) 5

ADD UPTO THE NUMBER FROM A SINGLE NUMBER

Create a function that takes a number as an argument. Add up all the numbers from 1 to the number you passed to the function. For example, if the input is 4 then your function should return 10 because 1+2+3+4 = 10

ANY PRIME NUMBER IN RANGE

Create a function that return true if there is at least one prime number in the given range(n1 to n2) inclusive, false otherwise.

Example:

primeInRange(10,15) ⟶ true

// prime number is range : 11, 13

primeInRange(3,1) ⟶ true

// prime number is range : 3, 5

GUESSING GAME

Generate a random number (do research) and store it in a variable. Write a program to take input from the user and tell them whether their guessed number is correct, greater or lesser than the original number. (100 – number of guesses) is the score of user. The program is expected to terminate once the number is guessed. Number should be between 1 – 100.

Example:

Random number generated by computer: 54

User input: 34

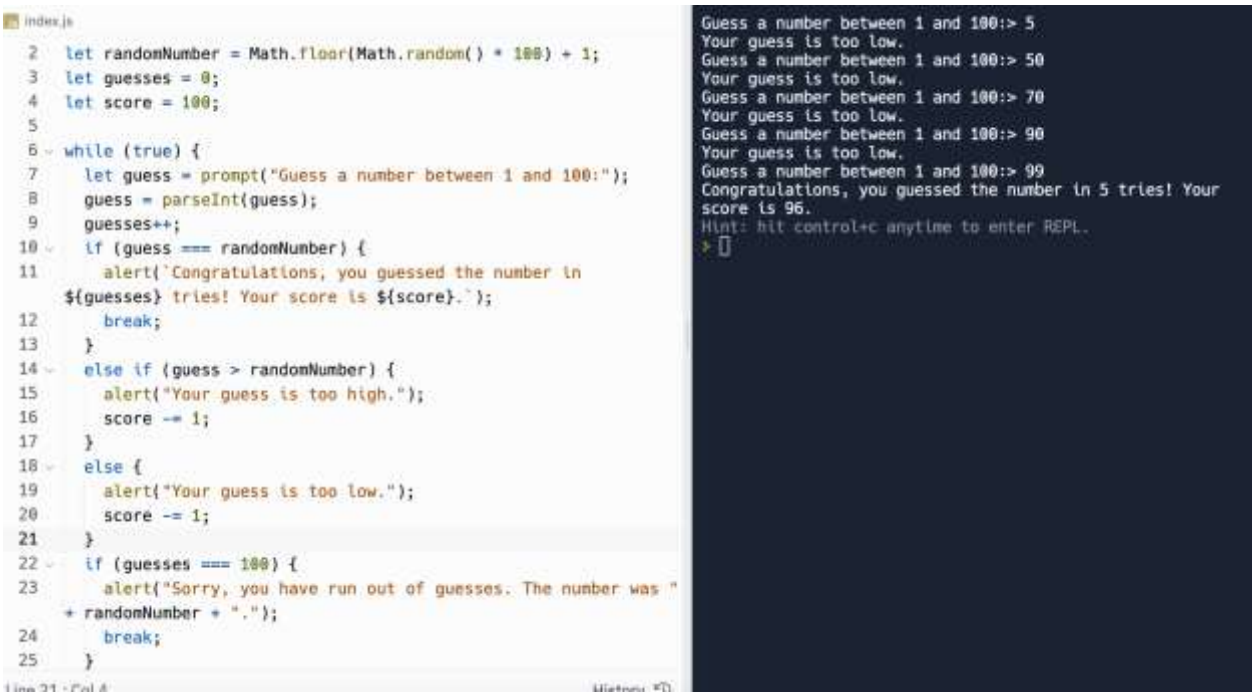
// lesser than original number

User input: 67

// greater than original number

User input: 54

// congratulations!!! The number you guessed matched the original number. Your score is 97!

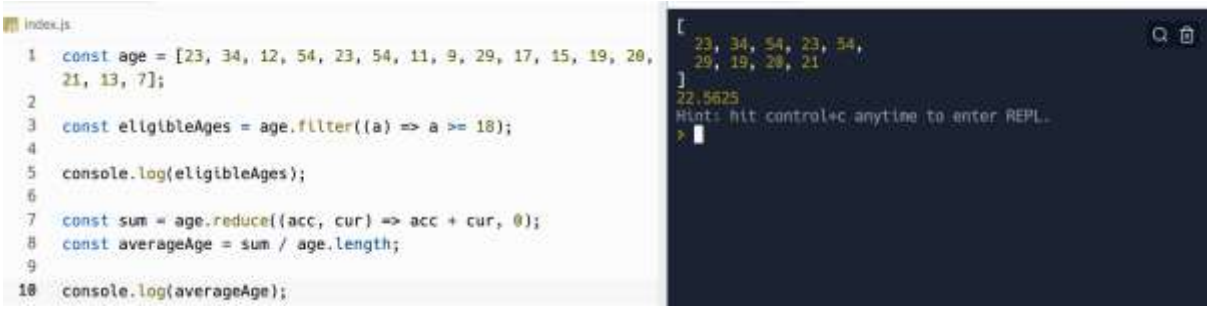


HIGHER ORDER ARRAY METHODS

Const age = [23,34,12,54,23,54,11,9,29,17,15,19,20,21,13,7]

a. Filter the array of age who can apply for citizenships

b. Find the average age of a given array



c. Const companies = [

{ name: "ABC", category: "Finance", start: 1981, end: 2004 },

{ name: "XYZ", category: "Retail", start: 1991, end: 20012 },

{ name: "DGF", category: "Finance", start: 1976, end: 2008 },

{ name: "LFT", category: "Retail", start: 1971, end: 1979 },

{ name: "MND", category: "Retail", start: 1995, end: 2010 },

{ name: "HCK", category: "Technology", start: 1987, end: 2011 },

{ name: "BMC", category: "Technology", start: 1989, end: 2009 },

{ name: "TIC", category: "Retail", start: 1993, end: 2005 },

{ name: "NAC", category: "Technology", start: 1991, end: 2010 },

{ name: "ITC", category: "Finance", start: 1998, end: 2016 }

];

index.js

```javascript
const companies = [
  { name: "ABC", category: "Finance", start: 1981, end: 2004 },
  { name: "XYZ", category: "Retail", start: 1991, end: 20012 },
  { name: "DGF", category: "Finance", start: 1976, end: 2008 },
  { name: "LFT", category: "Retail", start: 1971, end: 1979 },
  { name: "MND", category: "Retail", start: 1995, end: 2010 },
  { name: "HCK", category: "Technology", start: 1987, end: 2011 },
  { name: "BMC", category: "Technology", start: 1989, end: 2009 },
  { name: "TIC", category: "Retail", start: 1993, end: 2005 },
  { name: "NAC", category: "Technology", start: 1991, end: 2010 },
  { name: "ITC", category: "Finance", start: 1998, end: 2016 }
];
const retailCompanies = companies.filter(company => company.category === "Retail");
console.log(retailCompanies);
const eightiesCompanies = companies.filter(company => company.start >= 1980 && company.start < 1990);
console.log(eightiesCompanies);
const lastedTenYearsOrMore = companies.filter(company => (company.end - company.start) >= 10);
console.log(lastedTenYearsOrMore);
```

Line 19 : Col 1 History

Console Shell

```
[
  { name: 'XYZ', category: 'Retail', start: 1991, end: 20012 },
  { name: 'LFT', category: 'Retail', start: 1971, end: 1979 },
  { name: 'MND', category: 'Retail', start: 1995, end: 2010 },
  { name: 'TIC', category: 'Retail', start: 1993, end: 2005 }
]
[
  { name: 'ABC', category: 'Finance', start: 1981, end: 2004 },
  { name: 'HCK', category: 'Technology', start: 1987, end: 2011 },
  { name: 'BMC', category: 'Technology', start: 1989, end: 2009 }
]
[
  { name: 'ABC', category: 'Finance', start: 1981, end: 2004 },
  { name: 'XYZ', category: 'Retail', start: 1991, end: 20012 },
  { name: 'DGF', category: 'Finance', start: 1976, end: 2008 },
  { name: 'MND', category: 'Retail', start: 1995, end: 2010 },
  { name: 'HCK', category: 'Technology', start: 1987, end: 2011 },
  { name: 'BMC', category: 'Technology', start: 1989, end: 2009 },
  { name: 'TIC', category: 'Retail', start: 1993, end: 2005 },
  { name: 'NAC', category: 'Technology', start: 1991, end: 2010 },
  { name: 'ITC', category: 'Finance', start: 1998, end: 2016 }
```

a. Filter the retail companies
b. Get the 80s companies from the array
c. Get the companies that lasted for 10 or more years