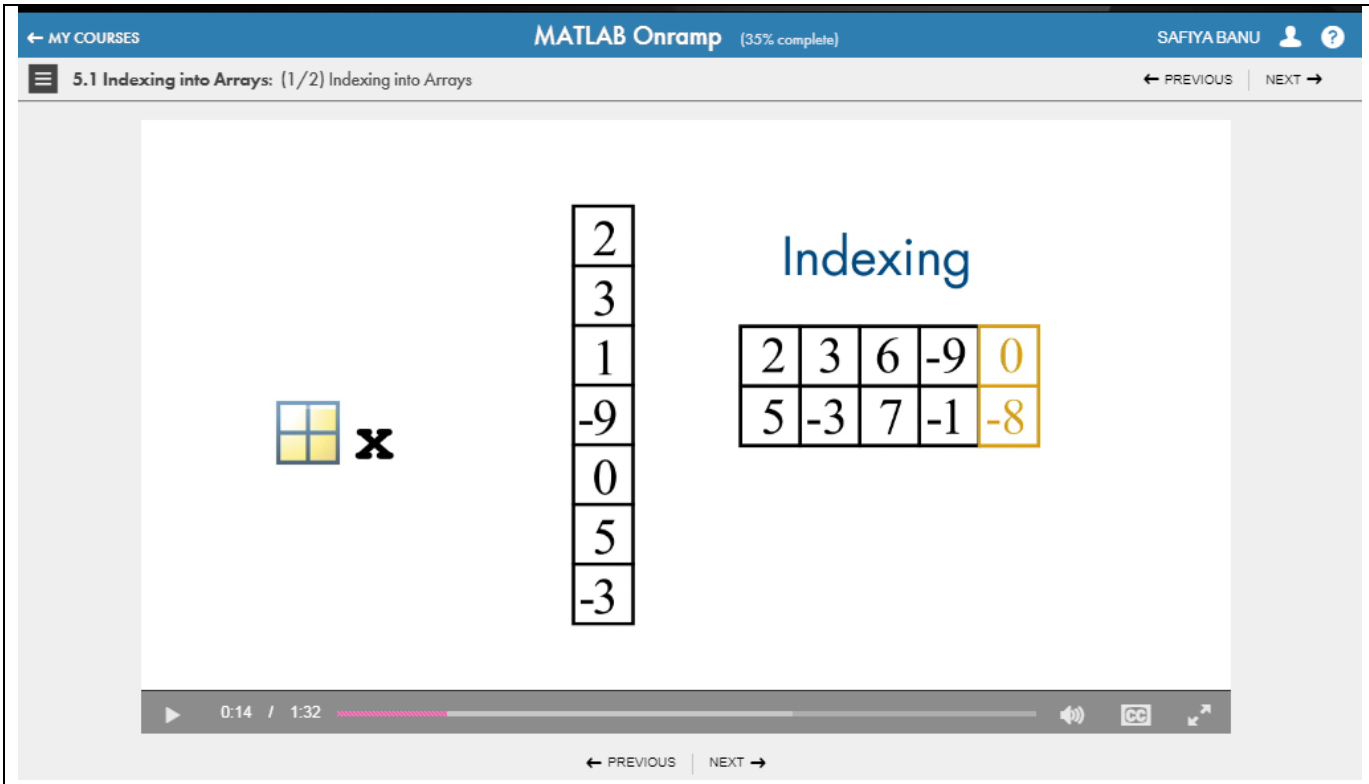
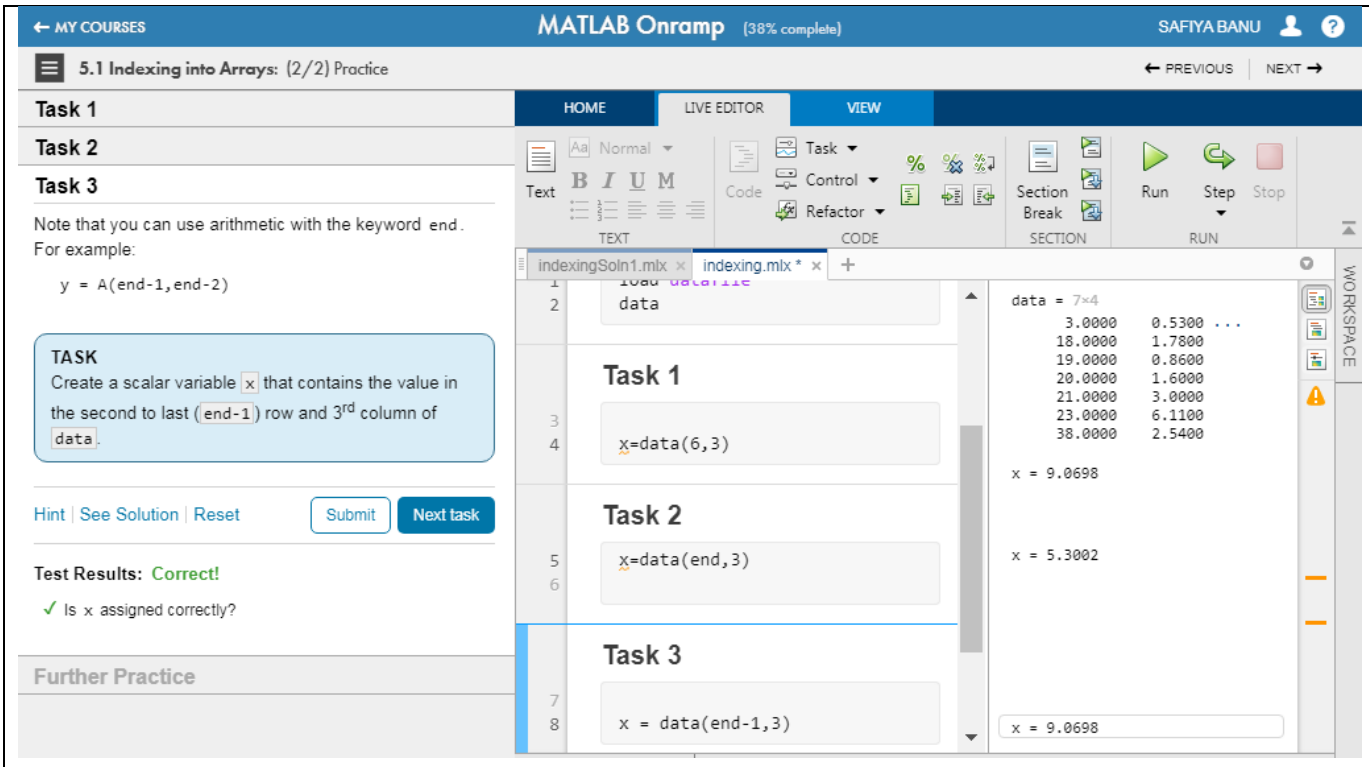
Report

Date:	**7July 2020**	**Name:**	**Safiya Banu**
Course:	**MATLAB Onramp**	**USN:**	**4AL16EC061**
Topic:	**1. Indexing into arrays** **i. Indexing into arrays** **ii. Extraxing multiple elements** **2. Array calculation** **I. Performingarray operations on vectors**	**Semester & Section:**	**8th sem "B"section**
Github Repository:	**Safiya-Courses**		





Extraxing multiple elements

When used as an index, the colon operator (:) specifies all the elements in that dimension. The syntax

x = A(2,:)

creates a row vector containing all of the elements from the second row of A.

TASK 1
Create a variable named density that contains the second column of the matrix named data.

load datafile
data
density=data(:,2)

Changing values in arrays

TASK 1
Create a vector named v2 containing the last column of data.

v2 = data(:,end)

TASK 2

Change the first element in v2 from NaN to 0.5.

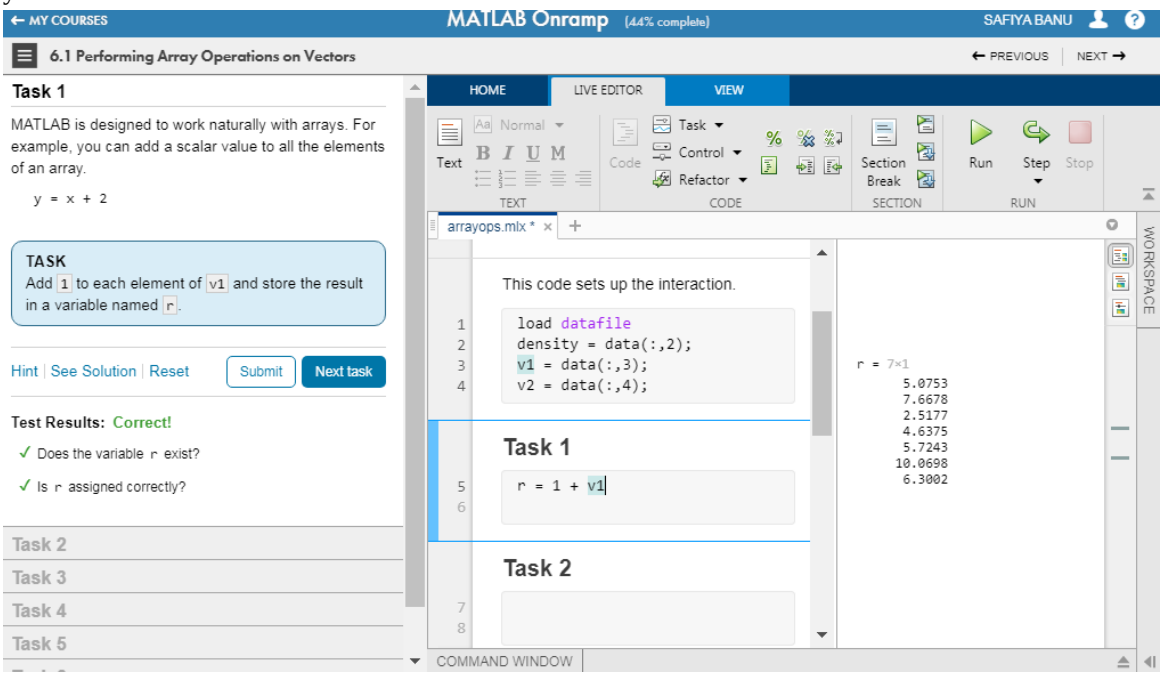
v2(1) = 0.5

ARRAY CALCULATIONS

Performing array operations on vectors

MATLAB is designed to work naturally with arrays. For example, you can add a scalar value to all the elements of an array.

y = x + 2



The * operator performs matrix multiplication. So, if you use * to multiply two equally sized vectors, since the inner dimensions do not agree, you will get an error message.

z = [3 4] * [10 20]

Error using *
Incorrect dimensions for matrix multiplication.

In contrast, the.* operator performs element wise multiplication and allows you to multiply the corresponding elements of two equally sized arrays.

z = [3 4] .* [10 20]

z = 30 80