# MODULE 17 : SUMMARY

* + - * + VARIABLES
        + JS PRINT PROCEDURE CONSOL/DOM
        + DATA TYPE
        + TYPE CONVERSIONS
        + OPERATORS

## JS OUTPUT : 17-1 Introduction to JavaScript, Run JavaScript in VSCode

* + CONSOLE
  + VS CODE TERMINAL
  + CODEPEN.IO

## VARIABLES : 17-2 What is variable, five things you need to declare a variable

* + RULES FOR VARIABLE-NAME DECLARE
    - PUT KEYWORD VAR / LET / CONST
    - PUT VARIABLE-NAME
    - PUT **=** SIGN
    - PUT VALUE
    - PUT **;** SIGN

## DARA-TYPE : 17-3 Variable type, Numeric, String, Boolean

* + Variable type
    - STRING
    - NUMBER
    - BOOLEAN

## 17-4 JavaScript Keyword, Variable name naming convention and best practice

* + VARIABLE NAMING RULES
    - RESERVED KEYWORDS
    - MEANINGFUL
    - NO SHORT-FORM FOR MULTI WORD VARIABLES
    - NO-SPACE FOR MULTI WORD VARIABLES
    - CAMEL-CASE FOR MULTI WORD VARIABLES

## 17-5 Simple Mathematical operations in JavaScript

* + OPERATORS : ADDITIONAL
    - PROJECT: RICE 40 TK; WATER 30TK ; RICE + WATER TK?
    - PROJECT: RICE 40 TK; WATER 30TK ; TOTAL TK?
    - PROJECT: ONE EGG 42 TK ; NEED 7 EGGS, HOW MANY TK?
    - PROJECT: ONE EGG 42 TK ; HAVE 700 TK, HOW MANY EGGS?

## 17-6 (advanced) Mathematical operation shorthand

* + RECAP of 17-5
  + OPERATORS : INCREMENT / DECREMENT
    - PROJECT: SHORTHAND INCREASE ONLY 1
    - PROJECT: SHORTHAND DECREASE ONLY 1
    - PROJECT: SHORTHAND INCREASE ONLY 10
    - PROJECT: SHORTHAND DECREASE ONLY 10

## DATA-TYPE CONVERSION: 17-7 (advanced) String Concatenation, Integer float parseInt parseFloat type conversion

* + OPERATORS : STRING
    - PROJECT : ADD ”ALI” WITH “RAHIM MUHAMAD”
  + NUMBER-TYPE : INTEGER AND FLOAT
    - PROJECT : CONVERT STRING “100” TO INTEGER-NUMBER 100
    - PROJECT : CONVERT STRING “10.25” TO FLOAT-NUMBER 10.25

## 17-8 VARIABLE TYPE AND USE OF typeOf, toFixed WITH parseFloat

* + FIND typeOf DIFFERENT DATA
    - NUMBER
    - STING
    - BOOLEAN
    - OBJECT
    - UNDEFINED
    - NULL
  + toFixed OPT

# MODULE 18 : SUMMARY

* + - * + ARRAY DATA-TYPE
        + ARRAY METHODS
        + CONDITIONS
        + CONDITIONS ON ARRAY.LENGTH
        + CONDITIONS ON ARRAY[VALUE-INDEX]
        + OPERATORS : COMPARISON

## 18-1 Module Introduction and Variable recap

* + Recap MODULE 17

## 18-2 Declare Array, array length and array index

* + DECLARING ARRAY-DATA
  + ARRAY METHODS
    - ARRAY,LENGTH

## 18-3 Array index, get and set by index, indexOf

* + ARRAY METHODS
    - ARRAY,INDEX
      * ARRAY.indexOf(INDEX-NO)
      * ARRAY.indexOf(INDEX-NO-OUT OF-CONTEXT)
      * ARRAY[INDEX-NO] = NEW-VALUE

## 18-4 Add or remove element from array using push, pop

* + ARRAY METHODS
    - ARRAY.push(NEW-VALUE);
    - ARRAY.pop();
    - NEWARRY = ARRAY.pop();

## 18-5 Compare variables and Comparison operator

* + OPERATORS :
    - LOGICAL
    - COMPARISON

## 18-6 Make conditional decision, if-else, comparison

* + CONDITIONS
    - IF
    - IF ELSE

## 18-7 Handle multiple conditions, and or

* + MULTIPLE CONDITIONS
    - AND / OR INSIDE IF

## 18-8 (advanced) Multi stage condition and nested conditions

* + NESTED CONDITIONS
    - IF ELSE-IF ELSE
    - IF INSIDE IF

## 18-9 Module summary and two more comparisons

* + RECAP MODULE 18

# MODULE 18-5 : SUMMARY

* + - * + WHILE LOOP
        + FOR LOOP
        + FOR LOOP ON ARRAY

## 18-5-1 Variable array and conditionals revision

* + Recap MODULE 18
    - VARIABLES
      * DATA TYPES
    - ARRAY
      * ARRAY METHODS
    - CONDITION
      * IF ELSE
      * IF ELSE IF ELSE

## 18-5-2 While loop, debug and understand while loop

* + WHILE LOOP
  + DEBUG LOOP WITH BREAK-POINTS IN VS CODE

## 18-5-3 More while loops, odd numbers, even numbers

* + WHILE LOOP
    - PROJECT : RUN A LOOP 1 - 10 WHERE INCREMENT STEPS 1
    - PROJECT : RUN A LOOP 1 - 100 WHERE OUTPUT WILL BE ALL EVEN NO
    - PROJECT : RUN A LOOP 1 - 100 WHERE OUTPUT WILL BE ALL ODD NO

## 18-5-4 For loop, how for loop works, while vs for loop

* + FOR LOOP
    - PROJECT : RUN A LOOP 1 - 10 WHERE INCREMENT STEPS 1
    - PROJECT : RUN A LOOP 1 - 100 WHERE OUTPUT WILL BE ALL EVEN NO
    - PROJECT : RUN A LOOP 1 - 100 WHERE OUTPUT WILL BE ALL ODD NO

## 18-5-5 Recap loop, run a loop for each element of an array

* + RECAP

# MODULE 19 : SUMMARY

* + - * + WHILE LOOP
        + FOR LOOP
        + FOR LOOP ON ARRAY

## 19-1 Module Introduction and concept Recap

* + RECAP LAST MODULES
    - VARIABLE
    - DATA TYPE
      * ARRAY
        + ARRAY METHODS
    - LOOP
      * WHILE LOOP
      * FOR LOOP
    - CONDITIONS

## 19-2 Declare a Function, call function, function vs loop

* + Declare Function

## 19-3 (advanced) Function parameter, function return

* + project f1 : 100 টাকা নিয়ে গিয়ে দোকানদার মামার কাছে পটেটো চাওয়ার function লিখতে হবে
  + project f2 : টাকা নিয়ে গিয়ে দোকানদার মামার কাছে পটেটো চাওয়ার function লিখতে হবে আর কয়টা পটেটো দিবে তাও জানতে হবে একপিচ ৫ টাকা

## 19-4 (advanced) Multiple parameter add, multiplication, etc

* + project f3 : add two number
  + project f4 : - two number
  + project f5 : / two number

## 19-5 Declare multiple objects with multiple properties

* + Declare object type data

## 19-6 multiple ways to get and set object property

* + Object methods
  + Edit object value

## 19-7 (optional) Javascript switch case break and default

* + Switch case a alt of if-else

## 19-8 (advanced) while and for loop break and continue

* + Break and continue in loop

## 19-9 Module summary and JS recap

* + Recap

# MODULE 20 : SUMMARY

* + - * + WHILE LOOP
        + FOR LOOP
        + FOR LOOP ON ARRAY

## 20-1 Module Introduction Apply JS and let, const

* + Recap
  + Introducing **let** and **const**

## 20-2 Unit Convert Inch to Feet, miles to kilometer

* + Multiple use of a single function
  + Project : convert inch to feet
  + Project : convert miles to km

## 20-3 Check even and odd number using function

* + Project : even and odd number

## 20-4 Check whether a year is a Leap Year or not

* + Project : find leapyear

## 20-5 Calculate Factorial of a number using for loop

* + Project : Factorial of 7 by loop
* 20-6 Recalculate factorial multiple times using a function
  + - Project : Factorial by Function