# **Assignments**

| 1. Accept a char input from the user and display it on the console. |
| --- |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *fmt.Println("Enter the input char:")*  *var character rune*  *\_, err := fmt.Scanf("%c", &character)*  *if err != nil {*  *fmt.Println("Error check")*  *}*  *fmt.Printf("The character entered is %c", character)*  *}* |
| 2. Accept two inputs from the user and output their sum.   | **Variable** | **Data Type** | | --- | --- | | Number 1 | Integer | | Number 2 | Float | | Sum | Float | |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *var num1 int*  *var num2 float64*  *fmt.Println("Enter the integer no:")*  *fmt.Scan(&num1)*  *fmt.Println("Enter the float no:")*  *fmt.Scan(&num2)*  *sum := float64(num1) + num2*  *fmt.Println("The sum of ", num1, " and ", num2, " is ", sum)*  *}* |
| 3. Write a program to find the simple interest.   * 1. Program should accept 3 inputs from the user and calculate simple interest for the given inputs. Formula: SI=(P\*R\*n)/100)  | **Variable** | **Data Type** | | --- | --- | | Principal amount (P) | Integer | | Interest rate (R) | Float | | Number of years (n) | Float | | Simple Interest (SI) | Float | |
| *package main*  *import "fmt"*  *func main() {*  *var p int*  *var r float64*  *var n float64*  *fmt.Println("Enter the Principal Amount")*  *fmt.Scan(&p)*  *fmt.Println("Enter the rate of Interest")*  *fmt.Scan(&r)*  *fmt.Println("Enter the no of years")*  *fmt.Scan(&n)*  *si := (float64(p) \* n \* r) / 100*  *fmt.Printf("The simple interest is %v", si)*  *}* |
| 4. Write a program to check whether a student has passed or failed in a subject after he or she enters their mark (pass mark for a subject is 50 out of 100).   1. Program should accept an input from the user and output a message as “Passed” or “Failed”  | **Variable** | **Data type** | | --- | --- | | mark | float | |
| *package main*  *import "fmt"*  *func main() {*  *var mark float64*  *fmt.Println("Enter the mark of the subject")*  *fmt.Scan(&mark)*  *if mark >= 50 && mark <= 100 {*  *fmt.Println("Passed")*  *} else if mark >= 0 && mark < 50 {*  *fmt.Println("Failed")*  *} else {*  *fmt.Println("Invalid mark entered")*  *}*  *}* |
| 5. Write a program to show the grade obtained by a student after he/she enters their total mark percentage.   1. Program should accept an input from the user and display their grade as follows  | **Mark** | **Grade** | | --- | --- | | > 90 | A | | 80-89 | B | | 70-79 | C | | 60-69 | D | | 50-59 | E | | < 50 | Failed |  | **Variable** | **Data type** | | --- | --- | | Total mark | float | |
| *package main*  *import "fmt"*  *func main() {*  *fmt.Println("Enter your mark: ")*  *var mark float64*  *fmt.Scan(&mark)*  *if mark >= 90 {*  *fmt.Println("Grade: A")*  *} else if mark >= 80 && mark <= 89 {*  *fmt.Println("Grade: B")*  *} else if mark >= 70 && mark <= 79 {*  *fmt.Println("Grade: C")*  *} else if mark >= 60 && mark <= 69 {*  *fmt.Println("Grade: D")*  *} else if mark >= 50 && mark <= 59 {*  *fmt.Println("Grade: E")*  *} else if mark >= 0 && mark <= 49 {*  *fmt.Println("Grade: F")*  *} else {*  *fmt.Println("Invalid mark entered")*  *}*  *}* |
| 6. Using the ‘switch case’ write a program to accept an input number from the user and output the day as follows.   | **Input** | **Output** | | --- | --- | | 1 | Sunday | | 2 | Monday | | 3 | Tuesday | | 4 | Wednesday | | 5 | Thursday | | 6 | Friday | | 7 | Saturday | | Any other input | Invalid Entry | |
| *package main*  *import "fmt"*  *func main() {*  *onceAgain:*  *fmt.Println("Enter the input no: 1-Sun, 2-Mon, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday")*  *var input int*  *fmt.Scan(&input)*  *switch input {*  *case 1:*  *fmt.Println("Sunday")*  *case 2:*  *fmt.Println("Monday")*  *case 3:*  *fmt.Println("Tuesday")*  *case 4:*  *fmt.Println("Wednesday")*  *case 5:*  *fmt.Println("Thursday")*  *case 6:*  *fmt.Println("Friday")*  *case 7:*  *fmt.Println("Saturday")*  *default:*  *fmt.Println("Invalid Entry, Enter the correct code once again")*  *goto onceAgain*  *}*  *}* |
| 7. Write a program to print the multiplication table of given numbers.   1. Accept an input from the user and display its multiplication table   Eg:  **Output**: Enter a number  **Input**: 5  **Output**:  1 x 5 = 5  2 x 5 = 10  3 x 5 = 15  4 x 5 = 20  5 x 5 = 25  6 x 5 = 30  7 x 5 = 35  8 x 5 = 40  9 x 5 = 45  10 x 5 = 50 |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *fmt.Println("Enter a number")*  *var input int*  *fmt.Scan(&input)*  *fmt.Println("The multiplication table of ", input, " is: ")*  *for i := 1; i <= 10; i++ {*  *fmt.Println(i, " \* ", input, " = ", i\*input)*  *}*  *}* |
| 8. Write a program to find the sum of all the odd numbers for a given limit   1. Program should accept an input as limit from the user and display the sum of all the odd numbers within that limit   For example if the input limit is 10 then the result is 1+3+5+7+9 = 25  **Output**: Enter a limit  **Input**: 10  **Output**: Sum of odd numbers = 25 |
| *package main*  *import "fmt"*  *func main() {*  *var limit int*  *fmt.Println("Enter the limit")*  *fmt.Scan(&limit)*  *var sum int = 0*  *for i := 1; i <= limit; i++ {*  *if i%2 != 0 {*  *sum = sum + i*  *}*  *}*  *fmt.Println("The sum of odd numbers upto ", limit, " is ", sum)*  *}* |
| 9. Write a program to print the following pattern (**hint**: use nested loop)  1  1 2  1 2 3  1 2 3 4  1 2 3 4 5 |
| *package main*  *import "fmt"*  *func main() {*  *for i := 1; i <= 5; i++ {*  *for j := 1; j <= i; j++ {*  *fmt.Print(" ", j, " ")*  *}*  *fmt.Println()*  *fmt.Println()*  *}*  *}* |
| 10. Write a program to interchange the values of two arrays.   1. Program should accept an array from the user, swap the values of two arrays and display it on the console   Eg: **Output**: Enter the size of arrays  **Input**: 5  **Output**: Enter the values of Array 1  **Input**: 10, 20, 30, 40, 50  **Output**: Enter the values of Array 2  **Input**: 15, 25, 35, 45, 55  **Output**: Arrays after swapping:  Array1: 15, 25, 35, 45, 55  Array2: 10, 20, 30, 40, 50 |
| *package main*  *import "fmt"*  *func main() {*  *fmt.Println("Enter the size of the arrays")*  *var size int*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of array 1")*  *var array1 = make([]int, size)*  *arrayIn(array1, size)*  *fmt.Println("Enter the values of array 2")*  *var array2 = make([]int, size)*  *arrayIn(array2, size)*  *fmt.Println("Array 1 entered is: ", array1)*  *fmt.Println("Array 2 entered is: ", array2)*  *temp := array1*  *array1 = array2*  *array2 = temp*  *fmt.Println("The Arrays after swapping:")*  *fmt.Println("Array 1: ", array1)*  *fmt.Println("Array 2: ", array2)*  *}*  *func arrayIn(array []int, len int) {*  *for i := 0; i < len; i++ {*  *fmt.Scan(&array[i])*  *}*  *}* |
| 11. Write a program to find the number of even numbers in an array   1. Program should accept an array and display the number of even numbers contained in that array   Eg: **Output**: Enter the size of an array  **Input**: 5  **Output:** Enter the values of array  **Input:** 11, 20, 34, 50, 33  **Output:** Number of even numbers in the given array is 3 |
| *package main*  *import "fmt"*  *func main() {*  *var size int*  *fmt.Println("Enter the size of the Array: ")*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of the array:")*  *var array = make([]int, size)*  *for i := 0; i < size; i++ {*  *fmt.Scan(&array[i])*  *}*  *flag := 0*  *for i := 0; i < size; i++ {*  *if array[i]%2 == 0 {*  *flag++*  *}*  *}*  */\* Another method*  *for \_, val := range array {*  *if val%2 == 0 {*  *flag++*  *}*  *}*  *\*/*  *fmt.Println("The no of even numbers in the array is: ", flag)*  *}* |
| 12. Write a program to sort an array in descending order   1. Program should accept and array, sort the array values in descending order and display it   Eg: **Output**: Enter the size of an array  **Input**: 5  **Output**: Enter the values of array  **Input**: 20, 10, 50, 30, 40  **Output**: Sorted array:  50, 40, 30, 20, 10 |
| *package main*  *import (*  *"fmt"*  *"sort"*  *)*  *func main() {*  *var size int*  *fmt.Println("Enter the size of the array")*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of the array")*  *var array = make([]int, size)*  *for i := 0; i < size; i++ {*  *fmt.Scan(&array[i])*  *}*  *fmt.Println("The entered array is: ", array)*  *//sort.Ints(array) //sorting in ascending order*  *// fmt.Println(array)*  *sort.Sort(sort.Reverse(sort.IntSlice(array)))*  *fmt.Println("The sorted array in descending Order is: ", array)*  *}* |
| 13. Write a program to identify whether a string is a palindrome or not   1. A string is a palindrome if it reads the same backward or forward eg: MALAYALAM   Program should accept a string and display whether the string is a palindrome or not  Eg: **Output**: Enter a string  **Input**: MALAYALAM  **Output**: Entered string is a palindrome  Eg 2: **Output**: Enter a string  **Input**: HELLO  **Output**: Entered string is not a palindrome |
| *package main*  *import "fmt"*  *func main() {*  *var str string*  *fmt.Println("Enter a string")*  *fmt.Scan(&str)*  *var flag int*  *for i := 0; i < len(str)/2; i++ {*  *if str[i] != str[len(str)-1-i] {*  *flag = 1*  *break*  *}*  *}*  *if flag == 1 {*  *fmt.Println("Entered string is Not a Palindrome")*  *} else {*  *fmt.Println("Entered string is Palindrome")*  *}*  *}* |
| 14. Write a program to add to two dimensional arrays   1. Program should accept two 2D arrays and display its sum   Eg: **Output**: Enter the size of arrays  **Input**: 3  **Output**: Enter the values of array 1  **Input**:  1 2 3  4 5 6  7 8 9  **Output**: Enter the values of array 2  **Input**:  10 20 30  40 50 60  70 80 90  **Output**: Sum of 2 arrays is:  11 22 33  44 55 66  77 88 99 |
| *package main*  *import "fmt"*  *func main() {*  *var size, value int*  *fmt.Println("Enter the size of Arrays")*  *fmt.Scan(&size)*  *var slice1 [][]int*  *var slice2 [][]int*  *var sumSlice [][]int*  *var tempSlice []int*  *fmt.Println("Enter the values of first array")*  *for i := 0; i < size; i++ {*  *for j := 0; j < size; j++ {*  *fmt.Scan(&value)*  *tempSlice = append(tempSlice, value)*  *}*  *slice1 = append(slice1, tempSlice)*  *tempSlice = nil*  *}*  *fmt.Println("Enter the values of second array")*  *for i := 0; i < size; i++ {*  *for j := 0; j < size; j++ {*  *fmt.Scan(&value)*  *tempSlice = append(tempSlice, value)*  *}*  *slice2 = append(slice2, tempSlice)*  *tempSlice = nil*  *}*  *fmt.Println("Array 1 is:\n", slice1)*  *fmt.Println("Array 2 is:\n", slice2)*  *for i := 0; i < size; i++ {*  *for j := 0; j < size; j++ {*  *value = slice1[i][j] + slice2[i][j]*  *tempSlice = append(tempSlice, value)*  *}*  *sumSlice = append(sumSlice, tempSlice)*  *tempSlice = nil*  *}*  *fmt.Println("The sum of arrays are:\n", sumSlice)*  *}* |
| 15. Write a program to accept an array and display it on the console using functions   1. Program should contain 3 functions including main() function   **main()**   1. Declare an array 2. Call function getArray() 3. Call function displayArray()   **getArray()**   1. Get values to the array   **displayArray()**   1. Display the array values |
| *package main*  *import "fmt"*  *func main() {*  *slice1 := getArray()*  *slice2 := getArray()*  *displayArray(slice1)*  *displayArray(slice2)*  *}*  *func getArray() []int {*  *var size, value int*  *var sliceInp []int*  *fmt.Println("Enter the size of the Array:")*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of the array")*  *for i := 0; i < size; i++ {*  *fmt.Scan(&value)*  *sliceInp = append(sliceInp, value)*  *}*  *return sliceInp*  *}*  */\* another method*  *func getArray() []int {*  *var size int*  *fmt.Println("Enter the size of the Array:")*  *fmt.Scan(&size)*  *sliceInp := make([]int, size)*  *fmt.Println("Enter the values of the array")*  *for i := 0; i < size; i++ {*  *fmt.Scan(&sliceInp[i])*  *}*  *return sliceInp*  *}*  *\*/*  *func displayArray(sliceOut []int) {*  *fmt.Println("The entered array is: \n", sliceOut)*  *}* |
| 16. Write a program to check whether a given number is prime or not   1. Program should accept an input from the user and display whether the number is prime or not   Eg: **Output**: Enter a number  **Input**: 7  **Output**: Entered number is a Prime number |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *var num int*  *fmt.Println("Enter a number:")*  *fmt.Scan(&num)*  *flag := 0*  *for i := 2; i < num; i++ {*  *if num%i == 0 {*  *flag++*  *break*  *}*  *}*  *if flag == 1 || num == 1 {*  *fmt.Println("The entered number ", num, " is Not a Prime number")*  *} else {*  *fmt.Println("The entered number ", num, " is a Prime number")*  *}*  *}* |
| 17. Write a menu driven program to do the basic mathematical operations such as addition, subtraction, multiplication and division (**hint**: use if else ladder or switch)   1. Program should have 4 functions named addition(), subtraction(), multiplication() and division() 2. Should create a class object and call the appropriate function as user prefers in the main function |
| *package main*  *import (*  *"fmt"*  *"os"*  *)*  *//just used structs*  *func main() {*  *var num1, num2, operation float64*  *var repeat string*  *fmt.Println("Enter the 2 numbers")*  *fmt.Scan(&num1)*  *fmt.Scan(&num2)*  *enteredValues := mathsOperation{num1, num2}*  *onceAgain:*  *fmt.Println("Enter the code for the operations to perform: \n 1 - Addition\n 2 - Substraction\n 3 - Multiplication\n 4 - Division\n 5 - Exit from the program")*  *fmt.Scan(&operation)*  *switch operation {*  *case 1:*  *enteredValues.addition()*  *case 2:*  *enteredValues.substraction()*  *case 3:*  *enteredValues.multiplication()*  *case 4:*  *enteredValues.division()*  *case 5:*  *os.Exit(0)*  *default:*  *fmt.Println("Wrong operation Entered")*  *}*  *fmt.Println("Press # to perform another operation or press any other key to to exit")*  *fmt.Scan(&repeat)*  *if repeat == "#" {*  *goto onceAgain*  *} else {*  *os.Exit(0)*  *}*  *}*  *type mathsOperation struct {*  *value1 float64*  *value2 float64*  *}*  *func (numbers mathsOperation) addition() {*  *sum := numbers.value1 + numbers.value2*  *fmt.Println("The result is ", sum)*  *}*  *func (numbers mathsOperation) substraction() {*  *sub := numbers.value1 - numbers.value2*  *fmt.Println("The result is ", sub)*  *}*  *func (numbers mathsOperation) multiplication() {*  *multiply := numbers.value1 \* numbers.value2*  *fmt.Println("The result is ", multiply)*  *}*  *func (numbers mathsOperation) division() {*  *divide := numbers.value1 / numbers.value2*  *fmt.Println("The result is ", divide)*  *}* |
| 18. Grades are computed using a weighted average. Suppose that the written test counts 70%, lab exams 20% and assignments 10%.  If Arun has a score of  Written test = 81  Lab exams = 68  Assignments = 92  Arun’s overall grade = (81x70)/100 + (68x20)/100 + (92x10)/100 = 79.5  Write a program to find the grade of a student during his academic year.   * 1. Program should accept the scores for written test, lab exams and assignments   2. Output the grade of a student (using weighted average)   Eg:  Enter the marks scored by the students  Written test = 55  Lab exams = 73  Assignments = 87  Grade of the student is 61.8 |
| *package main*  *import "fmt"*  *func main() {*  *var written, lab, assignment float64*  *fmt.Print("Enter the marks scored by the students:\nWritten Test:")*  *fmt.Scan(&written)*  *fmt.Print("Lab Exams:")*  *fmt.Scan(&lab)*  *fmt.Print("Assignments:")*  *fmt.Scan(&assignment)*  *grade(written, lab, assignment)*  *}*  *func grade(writ float64, lab float64, ass float64) {*  *studentGrade := (writ\*70)/100 + (lab\*20)/100 + (ass\*10)/100*  *fmt.Println("The overall grade of the student is ", studentGrade)*  *}* |
| 19. Income tax is calculated as per the following table   | **Annual Income** | **Tax percentage** | | --- | --- | | Up to 2.5 Lakhs | No Tax | | Above 2.5 Lakhs to 5 Lakhs | 5% | | Above 5 Lakhs to 10 Lakhs | 20% | | Above 10 Lakhs to 50 Lakhs | 30% |   Write a program to find out the income tax amount of a person.   1. Program should accept annual income of a person   Output the amount of tax he has to pay  Eg 1:  Enter the annual income  495000  Income tax amount = 24750.00  Eg 2:  Enter the annual income  500000  Income tax amount = 25000.00 |
| *package main*  *import "fmt"*  *func main() {*  *var income int*  *fmt.Println("Enter the annual income:")*  *fmt.Scan(&income)*  *if income <= 250000 {*  *fmt.Println("No tax to pay")*  *} else if income > 250000 && income <= 500000 {*  *tax := (income \* 5) / 100*  *fmt.Println("Income tax Amount = ", tax)*  *} else if income > 500000 && income <= 1000000 {*  *tax := (income \* 10) / 100*  *fmt.Println("Income tax Amount = ", tax)*  *} else if income > 100000 && income <= 5000000 {*  *tax := (income \* 30) / 100*  *fmt.Println("Income tax Amount = ", tax)*  *} else {*  *fmt.Println("Contact income tax dept.")*  *}*  *}* |
| 20. Write a program to print the following pattern using for loop  1  2 3  4 5 6  7 8 9 10 |
| *package main*  *import "fmt"*  *func main() {*  *k := 1*  *for i := 0; i < 4; i++ {*  *for j := 0; j <= i; j++ {*  *fmt.Print(k, " ")*  *k++*  *}*  *fmt.Println()*  *}*  *}* |
| 21. Write a program to multiply the adjacent values of an array and store it in an another array   * 1. Program should accept an array   2. Multiply the adjacent values   3. Store the result into another array   Eg:  Enter the array limit  5  Enter the values of array  1 2 3 4 5  Output  2 6 12 20 |
| *package main*  *import "fmt"*  *func main() {*  *var size, value int*  *var sliceArray []int*  *var resultSlice []int*  *fmt.Println("Enter the size of the array")*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of the array")*  *for i := 0; i < size; i++ {*  *fmt.Scan(&value)*  *sliceArray = append(sliceArray, value)*  *}*  *for i := 0; i < size-1; i++ {*  *resultSlice = append(resultSlice, sliceArray[i]\*sliceArray[i+1])*  *}*  *fmt.Println("The entered array is: ", sliceArray, "\nThe output array after multiply adjacent values is: ", resultSlice)*  *}* |
| 22. Write a program to add the values of two 2D arrays   1. Program should contains 3 functions including the main function   **main()**   1. Call function getArray() 2. Call function addArray() 3. Call function displayArray()   **getArray()**   1. Get values to the array   **getArray()**   1. Add array 1 and array 2   **displayArray()**   1. Display the array values   Eg:  Enter the size of array  2  Enter the values of array 1  1 2  3 4  Enter the values of array 2  5 6  7 8  Output:  Sum of array 1 and array 2:  6 8  10 12 |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *var size int*  *fmt.Println("Enter the size of the array")*  *fmt.Scan(&size)*  *fmt.Println("Enter the values of array 1")*  *slice1 := getArray(size)*  *fmt.Println("Enter the values of array 1")*  *slice2 := getArray(size)*  *resultSlice := sumArray(slice1, slice2, size)*  *fmt.Println("The input Array 1 enterd is: ")*  *displayArray(slice1)*  *fmt.Println("The input Array 2 entered is: ")*  *displayArray(slice2)*  *fmt.Println("The sum of Array1 and 2 is: ")*  *displayArray(resultSlice)*  *}*  *func getArray(length int) [][]int {*  *var value int*  *var tempSlice []int*  *var slice [][]int*  *for i := 0; i < length; i++ {*  *for j := 0; j < length; j++ {*  *fmt.Scan(&value)*  *tempSlice = append(tempSlice, value)*  *}*  *slice = append(slice, tempSlice)*  *tempSlice = nil*  *}*  *return slice*  *}*  *func sumArray(inpSlice1 [][]int, inpSlice2 [][]int, length int) [][]int {*  *var tempSlice []int*  *var addedSlice [][]int*  *for i := 0; i < length; i++ {*  *for j := 0; j < length; j++ {*  *tempSlice = append(tempSlice, inpSlice1[i][j]+inpSlice2[i][j])*  *}*  *addedSlice = append(addedSlice, tempSlice)*  *tempSlice = nil*  *}*  *return addedSlice*  *}*  *func displayArray(input [][]int) {*  *fmt.Println(input)*  *}* |
| 23. Write an object oriented program to store and display the values of a 2D array   * 1. Program should contains 3 functions including the main function   **main()**   1. Declare an array 2. Call function getArray() 3. Call function displayArray()   **getArray()**   1. Get values to the array   **displayArray()**   1. Display the array values   Eg:  Enter the size of array  3  Enter the array values  1 2 3  4 5 6  7 8 9  Array elements are:  1 2 3  4 5 6  7 8 9 |
| *package main*  *import (*  *"fmt"*  *)*  *func main() {*  *mySlice := myStructArray{}*  *// mySlice2 := myStructArray{}*  *// matrixvalues := mySlice2.getArray()*  *// mySlice2.displayArray(matrixvalues)*  *mySlice.displayArray(mySlice.getArray())*  *}*  *type myStructArray struct {*  *userSlice [][]int*  *size int*  *}*  *func (m \*myStructArray) getArray() [][]int {*  *var value int*  *var tempSlice []int*  *fmt.Println("Enter the size of the array")*  *fmt.Scan(&m.size)*  *fmt.Println("Enter the values of array")*  *for i := 0; i < m.size; i++ {*  *for j := 0; j < m.size; j++ {*  *fmt.Scan(&value)*  *tempSlice = append(tempSlice, value)*  *}*  *m.userSlice = append(m.userSlice, tempSlice)*  *tempSlice = nil*  *}*  *return m.userSlice*  *}*  *func (m myStructArray) displayArray(matrix [][]int) {*  *fmt.Println("The array is: ")*  *fmt.Println(matrix)*  *}* |
| 24. *String Compare function* |
| *// Go program to illustrate how to compare*  *// string using compare() function*  *package main*  *import (*  *"fmt"*  *"strings"*  *)*  *func main() {*  *var str1, str2 string*  *fmt.Println("Enter the first string:")*  *fmt.Scan(&str1)*  *fmt.Println("Enter the second string")*  *fmt.Scan(&str2)*  *if strings.Compare(str1, str2) == 0 {*  *fmt.Println("Both the strings are same")*  *} else {*  *fmt.Println("Both the strings are different")*  *}*  *}* |
| *25*  *\*\* GO Program to Swap Number Without Using Temporary Variables* |
| *package main*  *//swapping numbers without using third variable*  *package main*  *//swapping numbers without using third variable*  *import "fmt"*  *func main() {*  *var num1, num2 int*  *fmt.Println("Enter the first number")*  *fmt.Scan(&num1)*  *fmt.Println("Enter the second number")*  *fmt.Scan(&num2)*    *num1, num2 = num2, num1*  *fmt.Println("After swapping,\nNumber 1: ", num1, "\nNumber 2: ", num2)*  *}* |