

Profesor:

Neiner Maximiliano

Parte 1 - Ejercicios Simples

Aplicación Nº 1 (Sumar números)

Confeccionar un programa que sume todos los números enteros desde 1 mientras la suma no supere a 1000. Mostrar los números sumados y al finalizar el proceso indicar cuantos números se sumaron.

Aplicación Nº 2 (Cambiar color de fondo)

Realizar una aplicación con una funcionalidad sencilla, el color de fondo de la página cambia de color blanco al color rojo.

Nota: Utilizar la propiedad **background-color** del body.

Aplicación Nº 3 (Generar Tabla)

Utilizando objetos de *Html* (`<table>`, `<tr>`, `<td>`) se pide generar en forma dinámica una tabla. Dicha tabla se formará con 3 columnas y 5 filas.

Aplicación Nº 4 (Mostrar fecha y estación)

Obtenga la fecha actual del servidor y luego imprímala dentro de la página (Seleccione el formato que más le guste). Además indicar que estación del año es. Utilizar una estructura selectiva múltiple.

Parte 2 - Ejercicios con Formularios

Aplicación Nº 5 (Cambiar color de fondo II)

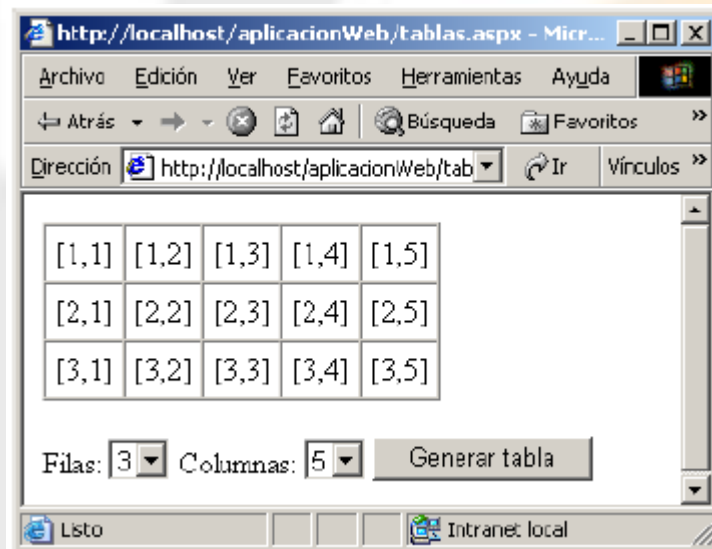
Realizar una aplicación que posea sólo un control *select*, con cinco o seis nombres de colores como opciones, y un control *input* (`type="button"`) con la leyenda "Cambiar Color". La funcionalidad de la aplicación es sencilla, se selecciona un color del select, se pulsa el botón y el color de fondo de la página cambia de color.

Nota: Utilizar la propiedad **background-color** del body.

Aplicación Nº 6 (Generar Tabla II)

Utilizando objetos de *Html* (`<table>`, `<tr>`, `<td>`) se pide generar en forma dinámica una tabla. Dicha tabla se formará a partir de los valores de dos controles de tipo *select*. Cada uno de estos controles contendrá valores desde el 1 hasta el 5.

Al pulsar el control *input* (`type="button"`) con la leyenda "Generar Tabla", se invocará a un procedimiento que creará la tabla a partir de la cantidad seleccionada de filas y columnas.



Aplicación Nº 7 (Superficie del cuadrado)

Confeccionar un formulario que solicite la medida del lado de un cuadrado. Luego mostrar la superficie del mismo:

- a- en la misma página.
- b- en otra página.

Aplicación Nº 8 (Superficie / Perímetro del cuadrado)

Modificar el formulario del ejercicio anterior para disponer de dos controles de tipo *radioButton* que permita seleccionar entre calcular la superficie y el perímetro del cuadrado.

El resultado se mostrará:

- a- en la misma página.
- b- en otra página.

Aplicación Nº 9 (Confirmar contraseña)

Solicitar el ingreso de una clave dos veces (es decir, disponer dos controles de tipo *password*), luego en el servidor verificar si las claves ingresadas son iguales o no, mostrando un mensaje de bienvenida, si son iguales o redireccionar a la página de inicio (que se deberá crear) en el caso de que sean distintos.

Aplicación Nº 10 (Tabla de Multiplicar)

Realizar una aplicación que lea un número entero de un *textBox* (txtNumero) y muestre en un *listBox* (lstTabla) la tabla de multiplicar de dicho número.

Aplicación Nº 11 (Mostrar fecha)

Realizar una página que permita mostrar el día, mes o año actual. Para esto debe utilizar controles de tipo *checkBox*.

Aplicación Nº 12 (Empresa de turismo)

Una empresa de turismo ofrece cinco destinos: *Río de Janeiro, Punta del Este, La Habana, Miami e Ibiza*. Se pide hacer una página que posea un *select* con los cinco destinos y un botón que le permita al usuario ver el valor del viaje.

Los valores de los viajes son: 900, 550, 1000, 1250 y 1500 respectivamente.

Aplicación Nº 13 (Empresa de turismo con promociones)

Modificar la aplicación anterior para que la empresa pueda ofrecer una promoción de acuerdo al modo de pago y la cantidad de pasajes a comprar.

Si el pago es en efectivo se realizará un descuento del 12% del valor del pasaje. Si es por medio de tarjetas de crédito o débito el descuento será del 7%.

Independientemente de la forma de pago si la cantidad de pasajes es superior a 2 cada pasaje extra se abonará el 35% menos.

Aplicación Nº 14 (Solicitud de empleo)

Confeccionar un formulario que permita ingresar en una serie de controles de tipo *textBox* el nombre y apellido de una persona, su edad, su dirección, su mail y en un control de tipo *textArea* su currículum. Imprimir los datos cargados en una página PHP.

Aplicación Nº 15 (Descuento por Compra)

Un restaurante ofrece un descuento del 10% para consumos entre \$ 90 y \$ 150 y un descuento del 20% para consumos mayores a \$ 150. Para todos los demás casos no se aplica ningún tipo de descuento.

Elaborar una aplicación web que permita determinar el importe a pagar por el consumidor.

Aplicación Nº 16 (Información del Número)

Construya una aplicación que permita el ingreso de un número entero y muestre en pantalla la siguiente información:

- 1) Cantidad de cifras
- 2) Suma de cifras impares

- 3) Suma de cifras pares
- 4) Suma total de cifras
- 5) Divisores de dicho número.

El diseño de la interfaz debe ser similar a la figura siguiente:

Aplicación Nº 17 (Boleta de pago)

Se desea elaborar la boleta de pago de los trabajadores de una fábrica. Para ello se cuenta con los siguientes datos de entrada:

Apellidos y nombres de los trabajadores, sueldo básico y horas extras trabajadas.

Además, se sabe que los trabajadores trabajan en dos turnos: diurno y nocturno, y que la tarifa por las horas extras diurnas es de \$14 y por las horas extras nocturnas es de \$19. Las horas extras no pueden ser mayores a 20.

Los descuentos a aplicar son sobre el sueldo básico y se obtienen de la siguiente manera:

$$\text{Rentas} = \text{Sueldo Básico} * 0.11$$

$$\text{Fonavi} = \text{Sueldo Básico} * 0.06$$

$$\text{AFJP} = \text{Sueldo Básico} * 0.14$$

Finalmente el sueldo neto del trabajador se calcula mediante las siguientes expresiones:

$\text{Ingresos} = \text{Sueldo Básico} + \text{Cantidad de horas Extras} * \text{Pago por hora Extra}$

$\text{Egresos} = \text{Renta} + \text{Fonavi} + \text{AFJP}$

$\text{Sueldo Neto} = \text{Ingresos} - \text{Egresos}$

Se debe crear una clase llamada '**Boleta**' que contenga los siguientes atributos: Nombre, Apellido, Sueldo, Horas extras, Turno, Rentas, Fonavi, AFJP.

Todos los métodos estarán en la clase **Boleta**.

Utilizar **funciones JS de validación** sobre los campos clave.

Aplicación N° 18 (File UpLoad)

Se necesita crear una página que le permita al usuario subir al servidor web cualquier tipo de archivo. Solo se restringirá el tamaño de cada archivo según el tipo de extensión que posea.

Para archivos con extensión **.doc** o **.docx** el tamaño máximo será de **60 Kb**.

Archivos con extensión **.jpg**, **jpeg** o **gif** el valor máximo será de **300 kb**.

Para el resto de las extensiones el máximo permitido será de **500 kb**.

Dichos archivos se almacenarán en una carpeta llamada '**Uploads**' que se ubicará en el directorio raíz del servidor web.

Utilizar el control input type **File** y **funciones de validación**, que requiera que se seleccione un archivo antes de enviar la petición al servidor.

Se deberá informar si se logró subir el archivo o no. Si se pudo, informar el nombre del archivo, su extensión y que tamaño posee.

Parte 2 - Ejercicios con Bases de Datos

Aplicación N° 19 (Consulta de Notas)

Se necesita crear una aplicación web que les permita a los alumnos de la UTN consultar las notas obtenidas de cada materia.

La página de consulta (**ConsultaNota.php**) tendrá un *textBox*, en donde el alumno ingresará su DNI y una lista desplegable que contendrá todas las asignaturas.

Al pulsar el botón con la leyenda 'Consultar Nota' se visualizará el nombre y apellido del alumno con su correspondiente nota.

Utilizar una **base de datos**. Dicha base de datos tendrá tres tablas (Alumnos, Materias y Notas).

Tener en cuenta que el alumno pudo no haber cursado la materia a consultar.

Utilizar **funciones JS de validación**.

Aplicación N° 20 (Consulta de Notas II)

Partiendo de la aplicación del punto anterior, se necesita saber todas las notas del alumno que ingrese su número de documento o su nombre y apellido.

Esta información se enviará a otro formulario, dónde será procesada y se visualizará un listado completo de las materias y notas que posee el alumno.

Utilizar **funciones JS de validación**.

Utilizar **stored Procedures** para realizar las consultas.