# Contenido del curso.

1. **Introducción**.
   1. **Definición** **de Javascript**: Lenguaje de programación interpretado directamente por el navegador. No necesita ser compilado. Se usa para interactuar con la pagina web. Reproducir videos de youtube, ejecutar animaciones, validar formularios  
      Existen muchas librerías y frameworks para Javascript. Podemos crear aplicaciones móviles, aplicaciones de escritorio, administrar servidor
   2. Sería **recomendable aprender antes HTML que es un lenguaje de marcado que nos permitirá estructurar una pagina web y CSS que es un lenguaje para dar estilo a la página.**
   3. **Contiene los fundamentos esenciales de la programación y Javascript.**Aprenderás conceptos básicos como variables, tipos de datos, operadores, condicionales, funciones, métodos, ciclos y muchos otros temas importantes.
   4. **Enlace para descargar el código del curso:**<https://github.com/falconmasters/curso_javascript->
2. **Modulos ES6**. Son paquetes que pueden usarse en Javascript que contienen funciones, clases, vbles, que son importadas y pueden usarse en nuestro proyecto. Existen buscadores de paquetes. Nosotros usaremos npm que contiene el paquete rollup y la app git y su consola.
   1. Los paquetes son códigos de otras personas que podremos usar:
      1. Hay cientos de paquetes que podría buscar en <https://www.npmjs.com/> . Uno de ellos se llama react y se recomienda aprender despues de Javascript
      2. Para trabajar con npm necesitamos node.js. <https://nodejs.org/es.> Y node.js lo tiene instalado dentro
         1. Nodejs permite trabajar desde el lado del servidor. No lo veremos dentro de este curso
   2. **Git**
      1. https://git-scm.com/
      2. Permite trabajar con nuestro código en diferentes versiones
      3. Es muy útil cuando trabajamos en equipo
      4. En Linux y Mac viene preinstalado. En Windows no
      5. Proporciona una nueva consola con nuevos comandos que podrá integrarse con visual studio code. El siguiente vídeo explica como integrarse:  [https://youtu.be/EVh36sDppcs](https://youtu.be/EVh36sDppcs" \t "https://www.udemy.com/course/curso-javascript-desde-cero/learn/lecture/37780088" \l "questions/_blank)
      6. **Comandos en consola**
         1. clear. Limpia la pantalla
         2. cd carpeta. Entra en esa carpeta
         3. npx. Indica que es local. En el documento vienen ejemplos en globales. Debemos añadir npx si queremos hacerlo en local
         4. npm run <comando>. Hace correr un comando.
         5. Npm install. Si borré la carpeta node\_modules por error o deseo. La restaura dependiendo de la dependencia en package.json
   3. **Paquete Rollup**
      1. **Descripción**. Es una herramienta que permite compilar cada uno de los códigos que tenemos en cada uno de nuestros archivos de nuestro proyecto, generar un solo archivo que será al que haga referencia desde HTML. Toda la siguiente información ha sido obtenida desde <https://rollupjs.org/>
      2. **Preparación de carpetas**
         1. **Carpeta módulos** junto al proyecto
         2. **Carpeta src** dentro de módulos. Que contendrá:
            1. Archivos javascript del proyecto
            2. Archivo **index.js o main.js**. Archivo inicial donde importaremos código de otros archivos
         3. **Carpeta public**. Contendrá estilos, favicon, imágenes, el bundle, que contendrá todo el código al que llame desde html
      3. **Instalación**
         1. Abrir Visual studio code y a ver/terminal y mostrar la **consola git,** llamada gitbash
         2. Ir a la carpeta modulos mediante el comando **cd**
         3. Generar el **archivo package.json escribiendo el comando npm init -y**
         4. Instalar rollup de forma local o global escribiendo alguna de estas líneas de código obtenidas desde
            1. **Modo local: npm install rollup --save-dev**
            2. **Modo global: npm install --global rollup**
         5. **Probarlo para ver si funciona**
            1. Escribimos rollup para probarlo de forma global
            2. Escribimos npx rollup para probarlo de forma local
      4. **Uso de las sentencias export e import**
         1. Exporto contenido desde los archivos javascript origenes
         2. Importo dicho contenido desde main.js o index.
      5. **Ejecutar rollup**
         1. rollup - bandera <archivoACrear>
            1. Banderas:

-f cjs. Formato comun javascript

-o. Salida hacia..

-c. Compila todos los archivos del proyecto.

* + 1. **Compilación manual**
       1. **rollup src/index.js -f cjs**. Compila y muestra el resultado en consola
       2. **rollup src/index.js -o public/bundle.js -f cjs.** Compila y genera un archivo de salida.
    2. **Compilación automática.**
       1. Crear el archivo **rollup.config.js** en la carpeta raiz
          1. **Añadirle el código:** export default { input: 'src/index.js', output: { file: 'public\bundle.js', format: 'cjs' } };
       2. Abrimos **package.json**
          1. **Añadimos la línea: :** “build”: “rollup --config --watch” en los scripts. Aquí no ponemos npx si es local.
          2. Nos aseguramos que tenga la propiedad: “type”: “module”, al principio indicando que queremos trabajar con módulos.
          3. Ejecutamos npm run build

1. **Browser Objet Model (BOM**). Objeto propio del navegador que permite interactuar con él. **Objetos**:
   1. window
   2. location
   3. navigator
   4. alert
   5. timer
   6. cookie
2. **Document Object Model (DOM)**. Objeto para acceder a los elementos de una pagina web. Realizar sitios webs interactivos. Presionar un boton se realiza animación, efecto o ejecutar código.
3. **Primer proyecto: Galería de imágenes.**
   1. Portafolio de imagenes organizadas por categoría.
   2. La categoría contendrá un carrusel de imágenes por la que podremos movernos con un texto descriptivo de cada imagen.
4. **Segundo proyecto: Producto de una tienda.**
   1. Pagina de producto donde podrá seleccionar diferentes características, añadir cantidades y seleccionar entre diferentes pestañas donde se muestren características, reseñas y tipos de envío
   2. Seleccionar más de un producto
   3. Mostrar un carrito de compra con lo seleccionado anteriormente con la suma total del importe pudiéndose eliminar artículos del mismo
5. **Formularios.** Validación en tiempo real
6. **Tercer proyecto: Formulario para realizar transferencias**
   1. Validará los datos en tiempo real y no permitirá continuar hasta que no estén correctamente introducidos
   2. Antes de transferir se mostrarán los datos introducidos para que puedan ser confirmados antes de enviar la transferencia
7. **API’s.** Contectarse a servidores externos.
8. **Cuarto proyecto: Aplicación de películas**
   1. Conectarse a servidor de peliculas
   2. Filtros para mostrar resultados deseados
   3. Paginación para mostrarlos todos
9. **Quinto proyecto: Gastos personales**.
   1. CRUD de gastos con descripción.
   2. Cambio de monedas.
   3. Formatear fechas.
10. **Sexto proyecto: Portafolio de trabajos**
    1. Publicar mis trabajos
    2. Contendrá efectos
    3. Tendrá una galería con los trabajos
    4. Carrusel de comentarios de otras personas que podremos arrastrar por la pantalla para verlos todos