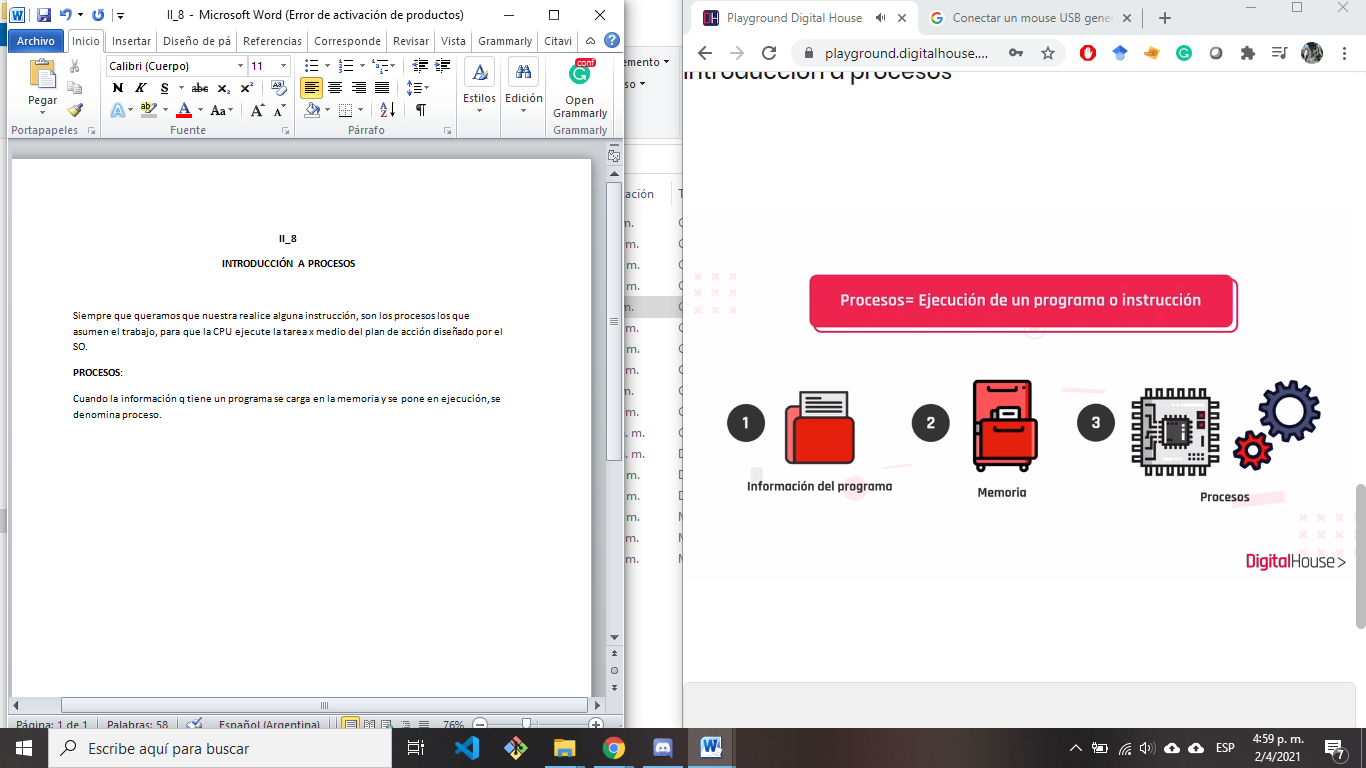
**II\_8**

**INTRODUCCIÓN A PROCESOS**

Siempre que queramos que nuestra realice alguna instrucción, son los procesos los que asumen el trabajo, para que la CPU ejecute la tarea x medio del plan de acción diseñado por el SO.

**PROCESOS**:

Cuando la información q tiene un programa se carga en la memoria y se pone en ejecución, se denomina proceso.



Todos los software ejecutables se organizan el procesos que quieren utilizar la CPU y es el SO que organiza el orden en que se van ejecutando esos procesos.

Este cambio de proceso se llama cambio de contexto.

Aunque creamos q todos los proceso se ejecutan al mismo tiempo, lo hacen sólo uno ala vez. No se almacenan en la memoria principal. TODOS LOS PROCESOS SON EFÍMEROS, SE CREAN Y SE ELIMINAN.

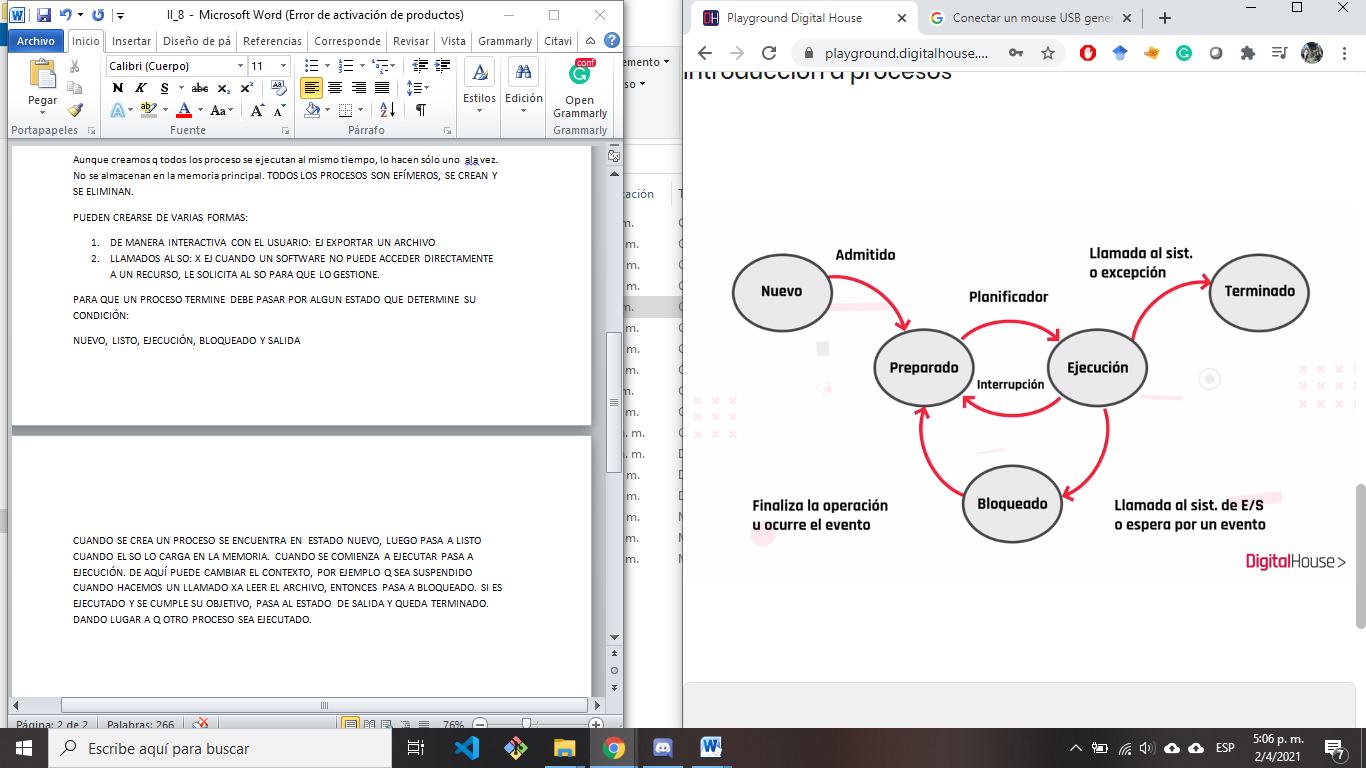
PUEDEN CREARSE DE VARIAS FORMAS:

1. DE MANERA INTERACTIVA CON EL USUARIO: EJ EXPORTAR UN ARCHIVO
2. LLAMADOS AL SO: X EJ CUANDO UN SOFTWARE NO PUEDE ACCEDER DIRECTAMENTE A UN RECURSO, LE SOLICITA AL SO PARA QUE LO GESTIONE.

PARA QUE UN PROCESO TERMINE DEBE PASAR POR ALGUN ESTADO QUE DETERMINE SU CONDICIÓN:

NUEVO, LISTO, EJECUCIÓN, BLOQUEADO Y SALIDA

CUANDO SE CREA UN PROCESO SE ENCUENTRA EN ESTADO NUEVO, LUEGO PASA A LISTO CUANDO EL SO LO CARGA EN LA MEMORIA. CUANDO SE COMIENZA A EJECUTAR PASA A EJECUCIÓN. DE AQUÍ PUEDE CAMBIAR EL CONTEXTO, POR EJEMPLO Q SEA SUSPENDIDO CUANDO HACEMOS UN LLAMADO XA LEER EL ARCHIVO, ENTONCES PASA A BLOQUEADO. SI ES EJECUTADO Y SE CUMPLE SU OBJETIVO, PASA AL ESTADO DE SALIDA Y QUEDA TERMINADO. DANDO LUGAR A Q OTRO PROCESO SEA EJECUTADO.



**EXISTEN VARIOS MECANISMOS DE COMUNICACIÓN ENTRE PROCESOS**

**IPC \_ INTER PROCESS**

**SEÑALES: AVISOS Q PUEDE ENVIAR UN P A OTRO.**

**MEMORIA COMPARTIDA: UN RECURSO COMPARTIDO A DISPOSICIÓN DE LOS SOFTWARE XA Q PUEDAN INTERCAMBIAR INFO**

**SE CREAN PROCESOS HIJOS Q REALIZAN SUBTAREAS XA Q EL PROCESO PADRE PUEDA CUMPLIR SU OBJETIVO. 1 PADRE PUEDE TENER VARIOS HIJOS.**

QUIZ

DUDA CAMBIO DE PROCESO = CAMBIO DE CONTEXTO

**QUIZ COMUNICACIÓN ENTRE PROCESOS**

[1](https://playground.digitalhouse.com/)

**[Existen dos tipos de procesos que se ejecutan de manera concurrente.](https://playground.digitalhouse.com/)**

Opción 1: Procesos independientes.

Opción 2: Procesos cooperativos.

**Corrección**  ¡Buen trabajo!

[2](https://playground.digitalhouse.com/)

**[En el modelo de memoria compartida se establece un espacio en memoria que será compartido por los procesos.](https://playground.digitalhouse.com/)**

Opción 1: Verdadero.

**Corrección**  ¡Buen trabajo!

[3](https://playground.digitalhouse.com/)

**[La memoria compartida es generalmente más \_\_\_\_\_\_\_\_\_\_\_ que usar un multiprocesador.](https://playground.digitalhouse.com/)**

Opción 2: Económica.

**Corrección**   ¡Buen trabajo!

[4](https://playground.digitalhouse.com/)

**[En el caso del modelo de pasos de mensajes, podemos decir que:](https://playground.digitalhouse.com/)**

Opción 2: La comunicación se hace mediante operaciones explícitas de envío y recepción.

Opción 1: Los procesos no comparten memoria.

**Corrección**   ¡Buen trabajo!

**QUIZ SINCRONIZACIÓN DE PROCESOS**

[1](https://playground.digitalhouse.com/)

**[La sincronización de procesos permite que mientras un proceso está escribiendo un registro, otro proceso no pueda leer.](https://playground.digitalhouse.com/)**

Opción 1: Verdadero.

**Corrección**  ¡Buen trabajo!

[2](https://playground.digitalhouse.com/)

**[Es una herramienta de sincronización](https://playground.digitalhouse.com/)**

Opción 2: Semáforo.

**Corrección**  ¡Buen trabajo!

[3](https://playground.digitalhouse.com/)

**[En esta planificación el procesador ejecuta los procesos por orden de llegada y los demás esperan en la cola.](https://playground.digitalhouse.com/)**

Opción 2: FIFO

**Corrección**  ¡Buen trabajo!

[4](https://playground.digitalhouse.com/)

**[Esta política de planificación toma de la cola de los procesos preparados el que necesite menos tiempo de ejecución para realizar su trabajo.](https://playground.digitalhouse.com/)**

Opción 3: SJF

**Corrección**  ¡Buen trabajo!

[5](https://playground.digitalhouse.com/)

**[Algunas características del proceso de planificación SRTF son:](https://playground.digitalhouse.com/)**

Opción 2: Puede ser injusta ya que un proceso corto puede echar a uno largo que esté haciendo uso del procesador y que además esté terminando.

Opción 3: Presenta mayor sobrecarga.

Opción 1: Es muy eficiente

**Corrección**  ¡Buen trabajo!

**QUIZ HILOS DE EJECUCIÓN**

[1](https://playground.digitalhouse.com/)

**[Un proceso puede dividirse en secuencias de tareas también denominadas:](https://playground.digitalhouse.com/)**

Opción 2: Hilos.

**Corrección**  ¡Buen trabajo!

[2](https://playground.digitalhouse.com/)

**[¿Podríamos decir que múltiples hilos pueden existir dentro de un proceso, ejecutándose de forma concurrente compartiendo recursos y memoria?](https://playground.digitalhouse.com/)**

Opción 1: Verdadero.

**Corrección**  ¡Buen trabajo!

[3](https://playground.digitalhouse.com/)

**[Los hilos, ¿forman parte física del procesador?](https://playground.digitalhouse.com/)**

Opción 2: Falso.

**Corrección**  ¡Buen trabajo!

[4](https://playground.digitalhouse.com/)

**[Algunas ventajas de los hijos sobre los procesos:](https://playground.digitalhouse.com/)**

Opción 2: Se tarda menos tiempo en crear un hilo nuevo en un proceso ya existente a crear un nuevo proceso.

**Corrección**  ¡Buen trabajo!

**CLASE 07/04/21**

**COLABORATIVOS: cuando un proceso termina una ejecución y guarda eso en memoria cuando otro le pide esa información para ejecutar el proceso.**

**Los procesos NO COMPARTEN RECURSOS. CUANDO LO UTILIZAN NO PUEDEN UTILIZAR ESOS RECURSOS NINGUN OTRO PROCESO.**

**LOS HILOS SI COMPARTEN RECURSOS PORQUE ESTÁN TRABAJNDO CON LOS RECURSOS ASIGNADOS A UN DETERMINADO PROCESO.**

**LOS PROCESOS INDEPENDIENTES NO PUEDEN SER AFECTADOS, PERO SI PUEDEN COMUNICARSE CON OTROS PROCESOS. - FALSO**

**PROCESOS INDEPENDIENTE TRABAJN SOLOS Y MODIFICAN EN MEMORIA, PERO NO SE COMUNICA CON OTROS PROCESOS, SI NO QUE LE PIDE AL KERNEL PARA PODER OCUPAR ESPACIOS DE MEMORIA.**

**UN PROCESO INDEPENDIENTE NO SE PUEDE COMUNICAR CON OTRO PROCESO.**

**PREGUNTAS PARCIAL**

1. TERMINAL

V O F - UN SCRIPT PUEDE ESTAR COMPUESTO POR UN UNICO COMANDO

VERDADERO

1. GITHUB

GIT REPO LOCAL Y GITHUB REPO REMOTO. DESPUES DE COMMIT DONDE NOS ENCONTRAMOS

EN EL LOCAL

1. PROCESOS

TODOS LOS SOFTWARES EJECUTABLES SE ORGANIZAN EN PROCESOS Q QUIEREN UTILIZAR LA CPU VOF

VERDADERO

1. COMO SE MIDE LA CAPACIDAD EN LAS MEMORIAS DUAL CHANNEL

SE SUMAN

1. V o F en un kernel monolítico, si el nucleo falla, se puede recuperar el sistema.

FALSO