

EVALUACIÓN	Obligatorio	GRUPO	TODOS	FECHA	14/09/2021
MATERIA	Programación para DevOps				
CARRERA	All - ASA				
CONDICIONES	<p>- Puntos: Máximo: 60 Mínimo:</p> <p>- Fecha máxima de entrega: 2/12/2021</p> <p>LA ENTREGA SE REALIZA EN FORMA ONLINE EN ARCHIVO NO MAYOR A 40MB EN FORMATO ZIP, RAR O PDF.</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none">- Inscribirse- Formar grupos de hasta tres personas.- Subir el trabajo a Gestión antes de la hora indicada, ver hoja al final del documento: <p>"RECORDATORIO"</p>				

Modalidad y puntaje

El obligatorio será hecho por dos o tres estudiantes (como mínimo 2 y como máximo 3 estudiantes).

Si bien es en grupo, la defensa es individual y la calificación también. Se consultará a cada participante, **por separado, sobre cualquier aspecto de la entrega del grupo.**

El obligatorio tiene un puntaje máximo de **50** puntos (esto a su vez es el 50% del puntaje total del curso) y se divide en dos ejercicios. El primero de **20** puntos (un script en bash) y el segundo de **30** puntos (un script desarrollado en Python).

Características generales

El objetivo básico de este obligatorio es hacer un script que liste información de los usuarios que se han conectado al sistema (suministrada por el comando **last**, utilizando solo los registros de ese comando que cumplen con ciertas condiciones de formato), permitiendo filtrar la información por un usuario pasado como parámetro y desplegar un reporte con el tiempo total de conexión, y hacer un script en Python que lo utilice apropiadamente para añadirle más funcionalidades y flexibilidad en la interpretación de sus parámetros, agregando opciones de filtrado de los campos de información a desplegar, ordenamiento de la información por un campo seleccionado, así como la posibilidad de invertir el orden del listado, y desplegar al final la cantidad total de registros encontrados.

El obligatorio consiste en la creación de un script en bash y un script en Python.

Ejercicio 1: Script en bash. Valor 20 puntos máximo.

Realizar un script en bash (shell script) que sea de la forma:

ej1_historial_usuarios_conectados.sh [-r] [-u Usuario]

El script deberá listar información del historial de usuarios conectados en el sistema, para lo cual utilizará el comando **last** apropiadamente (ese comando será su fuente de información). Los registros del comando **last** que no tengan correctamente un tiempo de finido de conexión, con la estricta forma "(hh:mm)" (horas y minutos, en dos dígitos), deberán ser descartados (como por ejemplo, los registros

correspondientes a usuarios que continúan conectados, o los que tienen más de un día de duración en la conexión). El script deberá mostrar en la primera línea los nombres de los campos que son desplegados. En caso de recibirse el modificador **-u** seguido de un nombre de usuario, la información a desplegar será solo de los registros de conexión de ese usuario. En caso de recibirse el modificador **-r**, se desplegará por la salida estándar, después del listado de conexiones y dejando una línea en blanco de separación, el tiempo total de conexión de las conexiones desplegadas (la suma de todos los tiempos de conexión de todos los registros listados) siguiendo el formato: **El tiempo total de conexión es: < Cantidad de días de tiempo de conexión > días, < Cantidad de horas de tiempo de conexión > horas y < Cantidad de minutos de tiempo de conexión > minutos**. Si la cantidad de tiempo de conexión no llega a un día (es menos de 24 horas en total), entonces no se incluirá el día en la salida del total de tiempo de conexión, quedando el formato: **El tiempo total de conexión es: < Cantidad de horas de tiempo de conexión > horas y < Cantidad de minutos de tiempo de conexión > minutos**. El orden de los registros desplegados debe ser el mismo que provea el comando **last** por defecto. Para obtener la información de las conexiones, el script de este ejercicio deberá usar solamente el comando **last** sin ningún tipo de modificadores.

En caso de recibirse el modificador **-u** y el mismo no sea seguido de un nombre de usuario, el script deberá desplegar el mensaje **"No se ha especificado el usuario para el modificador -u."** por su salida estándar de errores, devolviendo un 1 como código de retorno del script. Si el usuario ingresado seguidamente al modificador **-u** no existe en el sistema, el script deberá desplegar el mensaje **"No existe el usuario < usuario ingresado por el usuario > en el sistema."** por su salida estándar de errores, devolviendo un 2 como código de retorno del script.

Si el script recibe un número incorrecto de parámetros, se deberá desplegar un mensaje de error que diga **"Cantidad de parámetros errónea, solo se aceptan los modificadores -r y -u (seguido de un nombre de usuario)."** por la salida estándar de errores y se deberá devolver un 3 como resultado (como código de retorno del script).

Si se recibe un modificador inválido, se deberá desplegar el mensaje de error **"Modificador < modificador pasado como parámetro > incorrecto. Solo se aceptan -r y -u usuario, y en ese orden en caso de estar ambos presentes."** por la salida estándar de errores y se devolverá un 4 como código de retorno del script.

En caso de cualquier otro posible error (si es que se detectan más errores en el proceso de creación del script), se desplegará un mensaje de texto apropiado por la salida estándar de errores y se deberá retornar un 5 como resultado (como código de retorno del script).

Si no hay errores, se desplegará por la salida estándar las conexiones dadas por **last** (precedidas de los títulos de los campos) que cumplan los requerimientos (que tengan la duración de la conexión definida de la forma "(hh:mm)" y que pertenezcan al usuario indicado, si es que se ha ingresado el modificador **-u**), y se devolverá un 0 como código de retorno del script. Si tampoco se producen errores, pero no hay datos para listar (porque no hay un historial de conexiones que tengan claramente definida la duración de la conexión) y no se ha recibido el modificador **-u**, se desplegará por la salida estándar de errores el texto **"No se han encontrado conexiones para listar en el sistema."** Si se ha recibido el modificador **-u** con un usuario como parámetro y no se producen errores, pero no hay datos para listar (porque el usuario ingresado no tiene ninguna conexión que cumpla los requerimientos de formato de duración de la conexión), se desplegará por la salida estándar de errores el texto **"No se han encontrado conexiones para listar en el sistema para el usuario < usuario pasado como parámetro >."** En estos casos también se retornará un 0 como código de retorno del script (no se considera un error si no se han encontrado conexiones para listar,

sin importar si se ha ingresado o no el modificador **-u**). En este caso tampoco se desplegará el reporte de tiempo total de conexión aunque se haya recibido el modificador **-r**.

Si se recibe el modificador **-r**, después del listado de las conexiones (en caso que existan) y con un enter de separación, se desplegará el tiempo total de conexión en las condiciones antes descritas (suma de la duración de todas las conexiones listadas, con horas y minutos, o días, horas y minutos en caso que se superen las 24 horas de conexión).

Ejemplos de uso:

Suponiendo que no existe el usuario *momoch*, entonces, el comando:

ej1_historial_usuarios_conectados.sh -u momoch

Deberá desplegar, por su salida estándar de errores:

No existe el usuario momoch en el sistema.

Si el comando se ejecuta sin indicarle el usuario entonces, el comando:

ej1_historial_usuarios_conectados.sh -r -u

Deberá desplegar, por su salida estándar de errores:

No se ha especificado el usuario para el modificador -u.

o

suponiendo que el usuario *alumno* existe y tiene 6 registros como los mostrados en el listado siguiente (el comando *last*, filtrando solo las conexiones del usuario *alumno*, da lo siguiente):

<i>alumno</i>	<i>pts/0</i>	<i>:0</i>	<i>Sat Sep 4 14:24 - 14:25</i>	<i>(00:01)</i>
<i>alumno</i>	<i>pts/1</i>	<i>:0</i>	<i>Sat Sep 4 13:46 - 14:24</i>	<i>(00:37)</i>
<i>alumno</i>	<i>pts/0</i>	<i>:0</i>	<i>Sat Sep 4 12:55 - 14:18</i>	<i>(01:23)</i>
<i>alumno</i>	<i>:0</i>	<i>:0</i>	<i>Sat Sep 4 12:53 - 14:25</i>	<i>(01:31)</i>
<i>alumno</i>	<i>pts/0</i>	<i>:0</i>	<i>Wed Sep 1 17:03 - 17:03</i>	<i>(00:00)</i>
<i>alumno</i>	<i>:0</i>	<i>:0</i>	<i>Wed Sep 1 17:03 - crash</i>	<i>(1+22:54)</i>

entonces, la salida del siguiente comando:

ej1_historial_usuarios_conectados.sh -r -u alumno

Deberá desplegar (por su salida estándar):

<i>Usuario</i>	<i>Term</i>	<i>Host</i>	<i>Fecha</i>	<i>H. Con</i>	<i>H. Des</i>	<i>T. Con</i>
<i>alumno</i>	<i>pts/0</i>	<i>:0</i>	<i>Sat Sep 4 14:24 - 14:25</i>			<i>(00:01)</i>
<i>alumno</i>	<i>pts/1</i>	<i>:0</i>	<i>Sat Sep 4 13:46 - 14:24</i>			<i>(00:37)</i>
<i>alumno</i>	<i>pts/0</i>	<i>:0</i>	<i>Sat Sep 4 12:55 - 14:18</i>			<i>(01:23)</i>
<i>alumno</i>	<i>:0</i>	<i>:0</i>	<i>Sat Sep 4 12:53 - 14:25</i>			<i>(01:31)</i>

alumno pts/0 :0 Wed Sep 1 17:03 - 17:03 (00:00)

El tiempo total de conexión es: 3 horas y 31 minutos.

En el listado anterior se puede observar como aparecen solo **5** de los 6 registros originales. Esto es porque el último registro no tiene exactamente el formato de duración (hh:mm), ya que su duración ha sido de un día y 22 horas con 54 minutos, y por eso no debe ser tomado en cuenta. Se puede observar también los encabezados de los campos agregados por el script. Como se ha ingresado el modificador **-r**, al final se despliega el tiempo total de duración de las conexiones mostradas. Ese tiempo está solo en horas y minutos, y no en días, porque no se ha llegado a las 24 horas en total.

Ejercicio 2: script en Python. Valor 30 puntos máximo.

El objetivo del ejercicio 2 es hacer un script de Python que despliegue información producida en base a la salida del script creado en el ejercicio 1, agregándole funcionalidades y siendo más flexible en la interpretación de los modificadores (no requiriendo un orden específico para poder reconocerlos).

El ejercicio 2 consiste en realizar un script en Python que sea de la forma:

ej2_historial_usuarios_conectados_exp.py [-r] [-u Usuario] [-o {u,t,h,d}] [-i] [-f {u,t,h,f,c,n,d}]

El script de Python recibirá las indicaciones según lo ingresado por el usuario en la línea de comandos (modificadores, opciones y usuario) y desplegará información de conexiones del sistema en general o para el usuario indicado como parámetro. El script Python deberá aceptar los modificadores y sus parámetros en cualquier orden, pudiendo estar o no presentes.

Dos de los modificadores del script de Python (**-r** y **-u**) son iguales a los del ejercicio 1. Así, esas opciones serán tratadas de igual forma a ese script (se respetará, en la funcionalidad extendida del script en Python, que **-r** sea para desplegar el tiempo total de conexión de las conexiones listadas, así como que el modificador **-u** permita ingresar el usuario al cual se le desplegarán las conexiones).

A las opciones que tiene el script en bash del ejercicio 1, el script de Python le agregará más funcionalidades dadas por los siguientes modificadores:

Si el script en Python recibe el modificador **-h**, desplegará una ayuda (por la salida estándar) explicando para qué sirven los diferentes modificadores y la sintaxis general del comando.

Si el script en Python recibe el modificador **-o**, ordenará la salida por uno de cuatro posibles criterios en forma creciente. Si se recibe **u** como parámetro de la opción **-o** (el usuario ingresa en la línea de comandos **-o u**), entonces se ordenará el listado de conexiones por el nombre del usuario en forma alfabética creciente. Si se recibe **t** como parámetro de la opción **-o** (el usuario ingresa en la línea de comandos **-o t**), entonces se ordenará el listado de conexiones por la terminal en orden alfabético creciente. Si se recibe **h** como parámetro de la opción **-o** (el usuario ingresa en la línea de comandos **-o h**), entonces se ordenará el listado de conexiones por el host en orden alfabético creciente. Si se recibe **d** como parámetro de la opción **-o** (el usuario ingresa en la línea de comandos **-o d**), entonces se ordenará el listado de conexiones por la duración de la conexión en forma creciente (quedando de esta forma las conexiones que han durado menos primero y las que han durado más al final). Si se recibiera más de una opción de ordenamiento, se considerará una cualquiera de ellas (por ejemplo, la que se ingresó último en la línea de comandos al ejecutarse el script Python) y se ignorará la o las

demás. Si no se recibe el modificador **-o**, el orden de las conexiones será el mismo que retorne el script del ejercicio 1 (que a su vez respeta el orden que usa **last** por defecto).

Si el script en Python recibe el modificador **-i**, invertirá el orden del listado de salida, sea cual sea este. Es decir, este presente el modificador mencionado anteriormente (modificador **-o** y sus opciones **u**, **t**, **h** y **d**) o no, el modificador **-i** invertirá ese orden, por lo que se puede combinar con **-o** para producir, por ejemplo, un listado ordenado en forma decreciente por tiempo de duración de las conexiones (producido con los parámetros **-o d -i**), en vez de creciente.

Si el script en Python recibe el modificador **-f**, filtrará la salida quitando cualquiera de los 7 campos de las conexiones. Si se recibe **u** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f u**), entonces se filtrará (se quitara del listado) el campo que contiene al nombre de usuario. Si se recibe **t** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f t**), entonces se filtrará el campo que contiene a la terminal. Si se recibe **h** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f h**), entonces se filtrará el campo que contiene el host. Si se recibe **f** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f f**), entonces se filtrará el campo fecha. Si se recibe **c** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f c**), entonces se filtrará el campo que contiene la hora de conexión. Si se recibe **n** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f n**), entonces se filtrará el campo que contiene la hora de desconexión. Si se recibe **d** como parámetro de la opción **-f** (el usuario ingresa en la línea de comandos **-f d**), entonces se filtrará el campo que contiene la duración de la conexión. El script debe soportar el ingreso de más de una opción y proceder en consecuencia. Por ejemplo, si el usuario ingresa **-f h -f f** entonces se quitaran del listado los campos que contienen el host y la fecha. Si el usuario ingresa todas las opciones para filtrar todos los campos (y por tanto no se mostraría ningún listado por la salida estándar, ya que ningún campo sería visible), el script en Python deberá desplegar el mensaje **"Al menos un campo debe estar visible, no pudiéndose ocultar todos"** por la salida estándar de errores y se deberá devolver un 20 como resultado (como código de retorno del script). Observación: El script deberá ser capaz de ordenar la información por un campo que sea filtrado (procediendo a ordenar antes la información y recién después filtrar los campos no deseados). Por ejemplo, si el usuario ingresa los modificadores **-o d -f d**, deberá ordenar la información por duración en forma creciente aunque el campo duración de la conexión no sea mostrado (porque se ha filtrado por el modificador **-f** con parámetro **d**).

El script de Python deberá utilizar apropiadamente el script del ejercicio 1, no resolviendo por sí mismo (en ningún caso) directamente la obtención de la información de las conexiones de o los usuarios. El script en Python deberá llamar al script del ejercicio 1 con los parámetros adecuados para producir la salida a mostrar como resultado (haciendo el procesamiento necesario de la información brindada por el script del ejercicio 1 para adaptarla a los pedidos del usuario). Observe el paralelismo que hay entre dos de los modificadores (**-r** y **-u Usuario**) del script en Python con los modificadores del script del ejercicio 1.

El script en Python reutilizará la verificación de errores del script del ejercicio 1 (si se producen errores al ejecutarse ese script) y reutilizará también los mensajes que este envíe a su salida estándar de errores (los mensajes de error, que correspondan al ejercicio 1, deben ser generados por ese script, haciendo el script en Python un aprovechamiento de los mismos, reenviándolos a su propia salida estándar de errores). El código de retorno del script en Python será también el que retorne el script del ejercicio 1 en estos casos.

En caso que existan errores en la interpretación de los argumentos recibidos (cantidad incorrecta de argumentos, modificadores inválidos, parámetros incorrectos), se retornara un **25** como código de

retorno y se desplegará, por la salida estándar de errores, una descripción resumida de la sintaxis general del comando (del script en Python).

Si no se producen errores (con un usuario existente en el sistema en caso de usarse el modificador **-u** y no hay modificadores incorrectos ni otros errores en el ingreso de los argumentos), se desplegará la información que retorna el script del ejercicio 1 por su salida estándar, procesada y con las características apropiadas según los modificadores y parámetros indicados por el usuario. En este caso, el código de retorno deberá ser **0**.

Si al menos se ha listado una conexión, al final del listado y del reporte de tiempo total de conexión si se ha especificado el modificador **-r** (y con una línea de separación), el script en Python deberá desplegar por la salida estándar el texto "**Cantidad de conexiones listadas**": **< número de conexiones listadas >**" si no se ha recibido el modificador **-u**, y si se ha recibido el modificador **-u**, deberá desplegar "**Cantidad de conexiones listadas para el usuario < usuario ingresado >**": **< número de conexiones listadas >**.

Si no hay errores y además no hay conexiones para listar, también se aprovechará el mensaje que envía el script del ejercicio 1 en ese caso (por su salida estándar de errores), que será reenviado por la salida estándar de errores del script en Python. Como no es un error que no se encuentren conexiones en las condiciones indicadas (desde el punto de vista de la sintaxis de este script), el código de retorno del script de Python también será cero en este caso (como se da también con el script del ejercicio 1 en este mismo caso).

Si el script de Python necesita crear algún archivo auxiliar (es más recomendable no usar archivos auxiliares, pero está permitido utilizarlos), deberá hacerlo en el directorio */tmp* y asegurarse de que el archivo creado tenga un nombre único (alcanzando con agregarle al nombre del archivo el número de PID del script en Python, aunque se aceptarán soluciones más sofisticadas que eso). En caso de no poderse crear el archivo auxiliar en las condiciones necesarias, deberá desplegarse un mensaje de error adecuado por la salida estándar de errores y devolverse un **30** como resultado (como código de retorno del script). Antes de finalizar el script, el archivo o los archivos auxiliares creados deberán ser eliminados. Estas indicaciones son innecesarias si no se crean archivos auxiliares.

Por ejemplo, si se ejecuta:

ej2_historial_usuarios_conectados_exp.py -r -u alumno -o d -f t -f h -i

entonces la salida del script en Python deberá ser algo como (suponiendo las mismas condiciones mencionadas en el ejemplo del ejercicio 1):

<i>Usuario</i>	<i>Fecha</i>	<i>H. Con</i>	<i>H. Des</i>	<i>T. Con</i>
<i>alumno</i>	<i>Sat Sep</i>	<i>4 12:53</i>	<i>- 14:25</i>	<i>(01:31)</i>
<i>alumno</i>	<i>Sat Sep</i>	<i>4 12:55</i>	<i>- 14:18</i>	<i>(01:23)</i>
<i>alumno</i>	<i>Sat Sep</i>	<i>4 13:46</i>	<i>- 14:24</i>	<i>(00:37)</i>
<i>alumno</i>	<i>Sat Sep</i>	<i>4 14:24</i>	<i>- 14:25</i>	<i>(00:01)</i>
<i>alumno</i>	<i>Wed Sep</i>	<i>1 17:03</i>	<i>- 17:03</i>	<i>(00:00)</i>

El tiempo total de conexión es: 3 horas y 31 minutos.

Cantidad de conexiones listadas para el usuario alumno: 5.

Se puede observar de la anterior salida que se han eliminado del listado los campos de la terminal y el host debido a los modificadores y opciones **-f t -f h**. Puede observarse también que el listado ha sido ordenado en forma decreciente (por el modificador **-i**) de duración de la conexión (por el modificador **-o** con parámetro **d**). También se despliega el tiempo total de conexión por el modificador **-r**. Al final se despliega la cantidad de conexiones encontradas, que en este caso son 5.

Observaciones:

Los ejemplos mostrados en este texto son solo ilustrativos.

Elementos a entregar por cada grupo (en formato electrónico)

Se deberán entregar los siguientes **5** elementos en el informe a subir online (los que estarán en un único documento conteniendo a los **5**), siguiendo el procedimiento detallado en la última página de este documento, con título "**RECORDATORIO: IMPORTANTE PARA LA ENTREGA**". Además de estos **5** elementos que deberán ser entregarlos, se debe entregar la máquina virtual con la que se ha desarrollado el obligatorio y un directorio (fuera de la máquina virtual) conteniendo toda la estructura usada para el desarrollo del obligatorio. Los elementos a entregar deben estar en el siguiente orden:

Generales del obligatorio

- 1- Una página indicando **claramente** la siguiente información: los **nombres de los integrantes** del grupo (junto con sus **respectivas fotos**), sus números de estudiantes y los **caminos absolutos** donde se encuentra el script en bash del ejercicio 1 y el script en Python del ejercicio 2.

(1 elemento para la parte general del obligatorio).

Del ejercicio 1

- 2- El **código del script en bash del ejercicio 1**.
- 3- Un **ejemplo de la salida del script**, considerando un caso en que ha ejecutado sin errores.

(tenemos en total **2** elementos distintos para el ejercicio 1).

Del ejercicio 2

- 4- El **código del script en Python del ejercicio 2**.
- 5- Un **ejemplo de la salida del script en Python**, considerando un caso en que ha ejecutado sin errores.

(tenemos en total **2** elementos distintos para el ejercicio 2).

Observación: También debe ser entregada la máquina virtual en que se ha desarrollado el obligatorio.

OBSERVACIÓN:

En total, son **5 elementos distintos** a entregar en forma online entre los dos ejercicios y la información general. La pérdida mínima por cada elemento no entregado es de 5 puntos (pudiendo ser muchos más, como en el caso de faltar el **código de alguno de los scripts (en bash o en Python) en formato electrónico**, lo que implicaría la pérdida de al menos **20 puntos**), aunque se puedan obtener por otros medios o estén en un formato electrónico no entregado oportunamente, o se encuentren en la máquina virtual entregada. Tenga en cuenta también que se pierden puntos si no está disponible **toda la información** pedida en cada uno de esos **5 elementos distintos**.

Máquina virtual y directorio con archivos

Se entregará la máquina virtual donde se ha desarrollado el obligatorio (a coordinar con el grupo la forma de entrega, ya que no se puede subir online), que deberá contener todo lo pedido. La contraseña de **root** de la máquina virtual deberá ser la misma que se usa en el curso (**root01**). Además, se entregará un directorio conteniendo toda la estructura de archivos involucrada en la creación del obligatorio (código de los scripts, casos de prueba, ejemplos, directorios usados, etc.). Este directorio, y todo su contenido, no estará dentro de la máquina virtual (estará aparte de ella, debiendo comprimirse y adjuntarse a la entrega). Este directorio, comprimido, también se deberá entregar online en un archivo en formato zip o rar. Finalmente, en el único archivo zip o rar a entregar online, además de contener este **directorio** con lo pedido, estará también el documento (en pdf por ejemplo) con los **5 elementos descritos anteriormente**. Para la entrega de este archivo zip o rar, siga el procedimiento que se detalla en el apartado final **“RECORDATORIO: IMPORTANTE PARA LA ENTREGA”**.

Observación: EN CASO DE EXISTIR ALGUNA DISCREPANCIA ENTRE LA INFORMACIÓN ENTREGADA ONLINE Y LA DE LA MÁQUINA VIRTUAL EN EL MOMENTO DE LA DEFENSA, SE TOMARÁ COMO VALIDA LA QUE IMPLIQUE EL MENOR PUNTAJE PARA EL GRUPO, por lo que se recomienda tener la misma información entregada online para la defensa que la que hay en la máquina virtual, sin hacerle cambios, que solo pueden implicar pérdida de puntos y no un beneficio adicional.


Para entregar online los elementos solicitados en el formato electrónico (excepto la máquina virtual), se debe seguir el procedimiento detallado en el anexo “RECORDATORIO: IMPORTANTE PARA LA ENTREGA”, en la siguiente página.

RECORDATORIO: IMPORTANTE PARA LA ENTREGA

- **Obligatorios**

La entrega de los obligatorios será en formato digital online, a excepción de algunas materias que se entregarán en Bedelía y en ese caso recibirá información específica en el dictado de la misma.

Los principales aspectos a destacar sobre la **entrega online de obligatorios** son:

1. Ingresá al sistema de Gestión.
2. En el menú, seleccioná el ítem "Evaluaciones" y la instancia de evaluación correspondiente, que figura bajo el título "Inscripto".
3. Para iniciar la entrega hacé clic en el ícono: 
4. Ingresá el número de estudiante de cada uno de los integrantes y hacé clic en "Agregar". El sistema confirmará que los integrantes estén inscriptos al obligatorio y, de ser así, mostrará el nombre y la fotografía de cada uno de ellos. Una vez agregados todos los integrantes, hacé clic en "Crear equipo".

Cualquier integrante podrá:

- **Modificar la integración del equipo.**
- **Subir el archivo de la entrega.**

5. Seleccioná el archivo que deseás entregar. Verificá el nombre del archivo que aparecerá en la pantalla y hacé clic en "Subir" para iniciar la entrega. Cada equipo (hasta 2 estudiantes) debe entregar **un único archivo en formato zip o rar** (los documentos de texto deben ser pdf, y deben ir dentro del zip o rar). El archivo a subir debe tener **un tamaño máximo de 40mb**

Quando el archivo quede subido, se mostrará el nombre generado por el sistema (1), el tamaño y la fecha en que fue subido.

6. El sistema enviará un e-mail a todos los integrantes del equipo informando los detalles del archivo entregado y confirmando que la entrega fue realizada correctamente.
7. Podés cerrar la pestaña de entrega y continuar utilizando Gestión o salir del sistema.
8. La **hora tope para subir el archivo será las 21:00** del día fijado para la entrega.
9. La entrega se podrá realizar desde cualquier lugar (ej. hogar del estudiante, laboratorios de la Universidad, etc).
10. Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta **antes de las 20:00hs.** del día de la entrega, a través de los mails gervaz@ort.edu.uy, alamon@ort.edu.uy y terra@ort.edu.uy, o telefónicamente al 29021505 - int 1156 (de 8:00 a 14:00 hs) y 1436 (de 17:30 a 20:00 hs).

Si tuvieras una situación particular de fuerza mayor, debes contactarte con suficiente antelación al plazo de entrega, al Coordinador de Cursos (gervaz@ort.edu.uy) o Secretario Docente (paulos@ort.edu.uy).