
```
update(task_id, *, total=None, completed=None, advance=None, description=None, visible=None,
       refresh=False, **fields)
```

Update information associated with a task.

Parameters

- **task_id** (*TaskID*) – Task id (returned by `add_task`).
- **total** (*float*, *optional*) – Updates `task.total` if not None.
- **completed** (*float*, *optional*) – Updates `task.completed` if not None.
- **advance** (*float*, *optional*) – Add a value to `task.completed` if not None.
- **description** (*str*, *optional*) – Change task description if not None.
- **visible** (*bool*, *optional*) – Set visible flag if not None.
- **refresh** (*bool*) – Force a refresh of progress information. Default is False.
- ****fields** (*Any*) – Additional data fields required for rendering.

Return type

`None`

```
wrap_file(file, total=None, *, task_id=None, description='Reading...')
```

Track progress file reading from a binary file.

Parameters

- **file** (*BinaryIO*) – A file-like object opened in binary mode.
- **total** (*int*, *optional*) – Total number of bytes to read. This must be provided unless a task with a total is also given.
- **task_id** (*TaskID*) – Task to track. Default is new task.
- **description** (*str*, *optional*) – Description of task, if new task is created.

Returns

A readable file-like object in binary mode.

Return type

`BinaryIO`

Raises

`ValueError` – When no total value can be extracted from the arguments or the task.

```
class rich.progress.ProgressColumn(table_column=None)
```

Base class for a widget to use in progress display.

Parameters

`table_column` (*Optional[Column]*)

```
get_table_column()
```

Get a table column, used to build tasks table.

Return type

`Column`

```
abstractmethod render(task)
```

Should return a renderable object.

Parameters

`task` (`Task`)