

- **mode** (*str*) – The mode to use to open the file. Only supports “r”, “rb” or “rt”.
- **buffering** (*int*) – The buffering strategy to use, see `io.open()`.
- **encoding** (*str*, *optional*) – The encoding to use when reading in text mode, see `io.open()`.
- **errors** (*str*, *optional*) – The error handling strategy for decoding errors, see `io.open()`.
- **newline** (*str*, *optional*) – The strategy for handling newlines in text mode, see `io.open()`.
- **total** – (int, optional): Total number of bytes to read. Must be provided if reading from a file handle. Default for a path is `os.stat(file).st_size`.
- **description** (*str*, *optional*) – Description of task show next to progress bar. Defaults to “Reading”.
- **auto_refresh** (*bool*, *optional*) – Automatic refresh, disable to force a refresh after each iteration. Default is True.
- **transient** – (bool, optional): Clear the progress on exit. Defaults to False.
- **console** (`Console`, *optional*) – Console to write to. Default creates internal Console instance.
- **refresh_per_second** (*float*) – Number of times per second to refresh the progress information. Defaults to 10.
- **style** (*StyleType*, *optional*) – Style for the bar background. Defaults to “bar.back”.
- **complete_style** (*StyleType*, *optional*) – Style for the completed bar. Defaults to “bar.complete”.
- **finished_style** (*StyleType*, *optional*) – Style for a finished bar. Defaults to “bar.finished”.
- **pulse_style** (*StyleType*, *optional*) – Style for pulsing bars. Defaults to “bar.pulse”.
- **disable** (*bool*, *optional*) – Disable display of progress.
- **encoding** – The encoding to use when reading in text mode.

Returns

A context manager yielding a progress reader.

Return type

`ContextManager[BinaryIO]`

```
rich.progress.track(sequence, description='Working...', total=None, completed=0, auto_refresh=True,
                    console=None, transient=False, get_time=None, refresh_per_second=10,
                    style='bar.back', complete_style='bar.complete', finished_style='bar.finished',
                    pulse_style='bar.pulse', update_period=0.1, disable=False, show_speed=True)
```

Track progress by iterating over a sequence.

You can also track progress of an iterable, which might require that you additionally specify `total`.

Parameters

- **sequence** (*Iterable[ProgressType]*) – Values you wish to iterate over and track progress.
- **description** (*str*, *optional*) – Description of task show next to progress bar. Defaults to “Working”.