`no-unique-section-names`

This flag currently applies only to ELF-based targets using the LLVM codegen backend. It prevents the generation of unique ELF section names for each separate code and data item when `-Z function-sections` is also in use, which is the default for most targets. This option can reduce the size of object files, and depending on the linker, the final ELF binary as well.

For example, a function `func` will by default generate a code section called `.text.func`. Normally this is fine because the linker will merge all those `.text.*` sections into a single one in the binary. However, starting with [LLVM 12](#), the backend will also generate unique section names for exception handling, so you would see a section name of `.gcc_except_table.func` in the object file and potentially in the final ELF binary, which could add significant bloat to programs that contain many functions.

This flag instructs LLVM to use the same `.text` and `.gcc_except_table` section name for each function, and it is analogous to Clang's `-fno-unique-section-names` option.