

copyrighted material - no part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without permission in writing from the copyright owner.

- This check flags all array subscript expressions on static arrays and **std::arrays** that either do not have a constant integer expression index or are out of bounds (for **std::array**). For out-of-bounds checking of static arrays, see the **-Warray-bounds** Clang diagnostic.

- **Bounds.2**: Only index into arrays using constant expressions: Pass pointers to single objects (only) and Keep pointer arithmetic simple.

- Optionally, this check can generate fixes using **gsl::at** for indexing.

- If you provide clang-tidy with a path to **gsl/gsl.h** (non-standard)

4

0

cppcoreguidelines-pro-bounds-constant-array-index

- This check flags all array subscript expressions on static arrays and `std::arrays` that either do not have a constant integer expression index or are out of bounds (for `std::array`). For out-of-bounds checking of static arrays, see the `-Warray-bounds` Clang diagnostic.
- **Bounds.2:** Only index into arrays using constant expressions: Pass pointers to single objects (only) and Keep pointer arithmetic simple.
- Optionally, this check can generate fixes using `gsl::at` for indexing.
- If you provide clang-tidy with a path to `gsl/gsl.h` (non-standard)

```
clang-tidy -checks='cppcoreguidelines-pro-bounds-constant-array-index' file.cpp
```

```
int get_index();
```

```
int main()
```

```
{
```

```
    std::array arr = { 0, 1, 2, 3, 4, 5 };
```

```
    auto v = arr[get_index()];
```

```
}
```