

Content Block to Manual Test

Table of Contents

- [Table of Contents](#)
- [Happy Bullet Point Intro](#)
- [Development Topic 1](#)
- [Development Topic 2](#)

Happy Bullet Point Intro

THOSE PENGUINS ARE HAVIN' A GREAT TIME!

- A bullet of text. Like a slide!
 - How about a sub-bullet?
 - A bullet of text. Like a slide!
 - How about a sub-bullet?
 - A bullet of text. Like a slide!
 - How about a sub-bullet?

THOSE OTHER PENGUINS ARE HAVIN' A GREAT TIME!

- A bullet of text. Like a slide!
 - How about a sub-bullet?
 - A bullet of text. Like a slide!
- How about a sub-bullet?
- A bullet of text. Like a slide!



Development Topic 1

In a more traditional "user manual"-like presentation, you may want the freedom to escape from bullet point hell, while still keeping the topic focused - about a slide's-worth of content. So a column that would be stripped from the slide version.

In a more traditional "user manual"-like presentation, you may want the freedom to escape from bullet point hell, while still keeping the topic focused - about a slide's-worth of content. So a column that would be stripped from the slide version.

Test.cc

```
#include stuff.h

int goodbye(void) {

    int c << testme

    return 0

}
```

Line 1: Include some stuff. Make sure the environment variables are correct.

Line 7: That's all she wrote!

To compile this puppy:

```
rtems-gcc --flag --option \  
          --anotheroption \  
          <argument>
```

Always be sure to:

- De-allocate memory
- Motherhood statement 2
- And don't forget to de-allocate memory

Related Topics

- The minimal RCE application structure
- Required compiler flags

Reference Topics

- Compiler reference
- RCEAllocator class documentation
- RCEManager class documentation

Development Topic 2

Now, moving on to the next step, now vertically linear. Lorem ipsum.

Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum.

Test.cc

```
#include stuff.h

int goodbye(void) {

    int c << testme
    int d

    d = m.noreally_goodbye()

    return 0
}

int noreally_goodbye(void) {

    int x << testmeagain

    return 0
}
```

Lines 13-18: New function, noreally_goodbye

Line 6, 8: Allocate space for return value, call new function.



Beware calls in this context! Because:

- There
- Are
- Some
- Nasty
- Consequences...