

A cast to an unsized type was attempted.

Erroneous code example:

```
let x = &[1_usize, 2] as [usize]; // error: cast to unsized type: `[usize; 2]`  
                                //      as `[usize]`
```

In Rust, some types don't have a known size at compile-time. For example, in a slice type like `[u32]`, the number of elements is not known at compile-time and hence the overall size cannot be computed. As a result, such types can only be manipulated through a reference (e.g., `&T` or `&mut T`) or other pointer-type (e.g., `Box` or `Rc`). Try casting to a reference instead:

```
let x = &[1_usize, 2] as &[usize]; // ok!
```