

explicit_generic_args_with_impl_trait

The tracking issue for this feature is: #83701

The `explicit_generic_args_with_impl_trait` feature gate lets you specify generic arguments even when `impl Trait` is used in argument position.

A simple example is:

```
#![feature(explicit_generic_args_with_impl_trait)]

fn foo<T: ?Sized>(_f: impl AsRef<T>) {}

fn main() {
    foo::<str>("").to_string();
}
```

This is currently rejected:

```
error[E0632]: cannot provide explicit generic arguments when `impl Trait` is used in argument position
--> src/main.rs:6:11
   |
6 |     foo::<str>("").to_string();
   |           ^^^ explicit generic argument not allowed
```

However it would compile if `explicit_generic_args_with_impl_trait` is enabled.

Note that the synthetic type parameters from `impl Trait` are still implicit and you cannot explicitly specify these:

```
#![feature(explicit_generic_args_with_impl_trait)]

fn foo<T: ?Sized>(_f: impl AsRef<T>) {}
fn bar<T: ?Sized, F: AsRef<T>>(_f: F) {}

fn main() {
    bar::<str, _>("").to_string(); // Okay
    bar::<str, String>("").to_string(); // Okay

    foo::<str>("").to_string(); // Okay
    foo::<str, String>("").to_string(); // Error, you cannot specify `impl Trait` explicitly
}
```