

DEFINING A MONOID IN HASKELL

```
class Monoid a where
```

```
    mempty :: a
```

```
    (<>) :: a -> a -> a
```

DEFINING A MONOID IN HASKELL

```
class Monoid a where
```

```
    mempty :: a
```

```
    (<>) :: a → a → a
```

```
mconcat :: [a] → a
```

```
mconcat = foldr (<>) mempty
```