

```
# core.py
```

```
from db import db_read
```

```
def total_value(email, invoice_id, discount=0.0):  
    items = db_read(email, inv_no=invoice_id)  
    return sum(items)*(1.0-discount)
```

```
# test_core.py
```

```
from unittest.mock import patch, call
```

```
from core import total_value
```

```
def test_total_value():  
    with patch('core.db_read') as mock_read:  
        assert 300. == total_value("karl@python.de", 92)
```

```
# core.py
```

```
from db import db_read
```

```
def total_value(email, invoice_id, discount=0.0):  
    items = db_read(email, inv_no=invoice_id)  
    return sum(items)*(1.0-discount)
```

```
# test_core.py
```

```
from unittest.mock import patch, call
```

```
from core import total_value
```

```
def test_total_value():  
    with patch('core.db_read') as mock_read:  
        mock_read.return_value = [100, 200]  
        assert 300. == total_value("karl@python.de", 92)
```