

Pristatymo tema

Autoriai: Vardenis Pavardenis, Petras Petraitis

Turinys

1. Įvadas
2. Dėstymas
3. Išvados

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequale doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequale doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me.

Būti?

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequaleamur animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum.

Būti, ar nebūti?

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequaleamur animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum.

```
import sys
import signal
import warnings

from analyser.app import App
from analyser.qt_compat import QtWidgets
from analyser.settings import Settings

signal.signal(signal.SIGINT, signal.SIG_DFL)
warnings.filterwarnings("ignore", category=DeprecationWarning)

if __name__ == "__main__":
    settings = Settings()
    app = QtWidgets.QApplication(sys.argv)
    ex = App(settings)
    sys.excepthook = ex.excepthook
    sys.exit(app.exec_())
```

```
public interface SortedSet<E> extends Set<E> {  
    // Gražinamas aibės poaibis iki element.  
    Set<E> headSet(E element);  
    // Gražinamas aibės poaibis nuo element1 iki element2.  
    Set<E> subSet(E element1, E element2);  
    // Gražinamas aibės poaibis nuo element.  
    Set<E> tailSet(E element);  
    // Gražinamas atvirkštinis iteratorius.  
    Iterator<E> descendingIterator();  
}
```



```
public interface SortedSet<E> extends Set<E> {  
    // Gražinamas aibės poaibis iki element.  
    Set<E> headSet(E element);  
    // Gražinamas aibės poaibis nuo element1 iki element2.  
    Set<E> subSet(E element1, E element2);  
    // Gražinamas aibės poaibis nuo element.  
    Set<E> tailSet(E element);  
    // Gražinamas atvirkštinis iteratorius.  
    Iterator<E> descendingIterator();  
}
```

Ką tai reiškia?

