```
from SortAlgorithms import SortAlgorithms
import pandas as pd
import time
sorter_100 = SortAlgorithms(size=100, value_range=(1, 1000))
sorter_1000 = SortAlgorithms(size=1000, value_range=(1, 1000))
sorter_10000 = SortAlgorithms(size=10000, value_range=(1, 1000))
sorter_100000 = SortAlgorithms(size=100000, value_range=(1, 1000))


start_time = time.perf_counter()
sorter_100.mergeSort()
sorter_100._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_100.selectionSort()
sorter_100._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_100.bubbleSort()
sorter_100._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_100.improvedBubbleSort()
sorter_100._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_100.quickSort()
sorter_100._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_100.improvedQuickSort()
sorter_100._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_100.radixSort()
sorter_100._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)

# Export the dataframe to a CSV file
sorter_100.df.to_csv('results.csv', index=False)


start_time = time.perf_counter()
sorter_1000.mergeSort()
sorter_1000._log_to_dataframe('Merge Sort', time.perf_counter() -
```

```
start_time)

start_time = time.perf_counter()
sorter_1000.selectionSort()
sorter_1000._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_1000.bubbleSort()
sorter_1000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_1000.improvedBubbleSort()
sorter_1000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_1000.quickSort()
sorter_1000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_1000.improvedQuickSort()
sorter_1000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_1000.radixSort()
sorter_1000._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)

sorter_1000.df.to_csv('results.csv', mode='a', index=False,
header=False)



start_time = time.perf_counter()
sorter_10000.mergeSort()
sorter_10000._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_10000.selectionSort()
sorter_10000._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_10000.bubbleSort()
sorter_10000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)
```

```
start_time = time.perf_counter()
sorter_10000.improvedBubbleSort()
sorter_10000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_10000.quickSort()
sorter_10000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter_10000.improvedQuickSort()
sorter_10000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter_10000.radixSort()
sorter_10000._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)

sorter_10000.df.to_csv('results.csv', mode='a', index=False,
header=False)


sorter2_100 = SortAlgorithms(size=100, value_range=(1, 100))
sorter2_1000 = SortAlgorithms(size=1000, value_range=(1, 100))
sorter2_10000 = SortAlgorithms(size=10000, value_range=(1, 100))


start_time = time.perf_counter()
sorter2_100.mergeSort()
sorter2_100._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_100.selectionSort()
sorter2_100._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_100.bubbleSort()
sorter2_100._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_100.improvedBubbleSort()
sorter2_100._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
```

```
sorter2_100.quickSort()
sorter2_100._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_100.improvedQuickSort()
sorter2_100._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter2_100.radixSort()
sorter2_100._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)

# Export the dataframe to a CSV file
sorter2_100.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter2_1000.mergeSort()
sorter2_1000._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_1000.selectionSort()
sorter2_1000._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_1000.bubbleSort()
sorter2_1000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_1000.improvedBubbleSort()
sorter2_1000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter2_1000.quickSort()
sorter2_1000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_1000.improvedQuickSort()
sorter2_1000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
```

```
sorter2_1000.radixSort()
sorter2_1000._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)


sorter2_1000.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter2_10000.mergeSort()
sorter2_10000._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_10000.selectionSort()
sorter2_10000._log_to_dataframe('Selection Sort', time.perf_counter()
- start_time)

start_time = time.perf_counter()
sorter2_10000.bubbleSort()
sorter2_10000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_10000.improvedBubbleSort()
sorter2_10000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter2_10000.quickSort()
sorter2_10000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter2_10000.improvedQuickSort()
sorter2_10000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter2_10000.radixSort()
sorter2_10000._log_to_dataframe('Radix Sort', time.perf_counter() -
start_time)


sorter2_10000.df.to_csv('results.csv', mode='a', index=False,
header=False)

sorter3_100 = SortAlgorithms(size=100, value_range=(-1000, 10000))
sorter3_1000 = SortAlgorithms(size=1000, value_range=(-1000, 1000))
sorter3_10000 = SortAlgorithms(size=10000, value_range=(-1000, 1000))
```

```
start_time = time.perf_counter()
sorter3_100.mergeSort()
sorter3_100._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_100.selectionSort()
sorter3_100._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_100.bubbleSort()
sorter3_100._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_100.improvedBubbleSort()
sorter3_100._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter3_100.quickSort()
sorter3_100._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_100.improvedQuickSort()
sorter3_100._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

# Export the dataframe to a CSV file
sorter3_100.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter3_1000.mergeSort()
sorter3_1000._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_1000.selectionSort()
sorter3_1000._log_to_dataframe('Selection Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_1000.bubbleSort()
sorter3_1000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)
```

```
start_time = time.perf_counter()
sorter3_1000.improvedBubbleSort()
sorter3_1000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter3_1000.quickSort()
sorter3_1000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_1000.improvedQuickSort()
sorter3_1000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)

# Export the dataframe to a CSV file
sorter3_1000.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter3_10000.mergeSort()
sorter3_10000._log_to_dataframe('Merge Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_10000.selectionSort()
sorter3_10000._log_to_dataframe('Selection Sort', time.perf_counter()
- start_time)

start_time = time.perf_counter()
sorter3_10000.bubbleSort()
sorter3_10000._log_to_dataframe('Bubble Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_10000.improvedBubbleSort()
sorter3_10000._log_to_dataframe('Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter3_10000.quickSort()
sorter3_10000._log_to_dataframe('Quick Sort', time.perf_counter() -
start_time)

start_time = time.perf_counter()
sorter3_10000.improvedQuickSort()
sorter3_10000._log_to_dataframe('Improved Quick Sort',
time.perf_counter() - start_time)
```

```
# Export the dataframe to a CSV file
sorter3_10000.df.to_csv('results.csv', mode='a', index=False,
header=False)

sorter4_100 = SortAlgorithms(size=100, value_range=(1, 1000))
sorter4_100.mergeSort()
sorter4_100.original_array = sorter4_100.array
sorter4_1000 = SortAlgorithms(size=1000, value_range=(1, 1000))
sorter4_1000.mergeSort()
sorter4_1000.original_array = sorter4_100.array
sorter4_10000 = SortAlgorithms(size=10000, value_range=(1, 1000))
sorter4_10000.mergeSort()
sorter4_10000.original_array = sorter4_100.array


start_time = time.perf_counter()
sorter4_100.mergeSort()
sorter4_100._log_to_dataframe('Sorted Merge Sort', time.perf_counter()
- start_time)

start_time = time.perf_counter()
sorter4_100.selectionSort()
sorter4_100._log_to_dataframe('Sorted Selection Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_100.bubbleSort()
sorter4_100._log_to_dataframe('Sorted Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_100.improvedBubbleSort()
sorter4_100._log_to_dataframe('Sorted Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_100.quickSort()
sorter4_100._log_to_dataframe('Sorted Quick Sort', time.perf_counter()
- start_time)

start_time = time.perf_counter()
sorter4_100.improvedQuickSort()
sorter4_100._log_to_dataframe('Sorted Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_100.radixSort()
sorter4_100._log_to_dataframe('Sorted Radix Sort', time.perf_counter()
- start_time)
```

```
# Export the dataframe to a CSV file
sorter4_100.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter4_1000.mergeSort()
sorter4_1000._log_to_dataframe('Sorted Merge Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.selectionSort()
sorter4_1000._log_to_dataframe('Sorted Selection Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.bubbleSort()
sorter4_1000._log_to_dataframe('Sorted Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.improvedBubbleSort()
sorter4_1000._log_to_dataframe('Sorted Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.quickSort()
sorter4_1000._log_to_dataframe('Sorted Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.improvedQuickSort()
sorter4_1000._log_to_dataframe('Sorted Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_1000.radixSort()
sorter4_1000._log_to_dataframe('Sorted Radix Sort',
time.perf_counter() - start_time)

# Export the dataframe to a CSV file
sorter4_1000.df.to_csv('results.csv', mode='a', index=False,
header=False)


start_time = time.perf_counter()
sorter4_10000.mergeSort()
sorter4_10000._log_to_dataframe('Sorted Merge Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
```

```
sorter4_10000.selectionSort()
sorter4_10000._log_to_dataframe('Sorted Selection Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_10000.bubbleSort()
sorter4_10000._log_to_dataframe('Sorted Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_10000.improvedBubbleSort()
sorter4_10000._log_to_dataframe('Sorted Improved Bubble Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_10000.quickSort()
sorter4_10000._log_to_dataframe('Sorted Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_10000.improvedQuickSort()
sorter4_10000._log_to_dataframe('Sorted Improved Quick Sort',
time.perf_counter() - start_time)

start_time = time.perf_counter()
sorter4_10000.radixSort()
sorter4_10000._log_to_dataframe('Sorted Radix Sort',
time.perf_counter() - start_time)


# Export the dataframe to a CSV file
sorter4_10000.df.to_csv('results.csv', mode='a', index=False,
header=False)
```