Challenge: Implement Miner Interface

This lesson brings you a challenge to solve.

WE'LL COVER THE FOLLOWING

- Problem statement

Problem statement

Analogous to the Sorter interface, we developed previously, make a `Miner` interface with the necessary operations and a function `Min`, which has as a parameter a variable which is a collection of type `Miner` and which calculates and returns the minimum element in that collection.

Try to solve the challenge below. Good Luck!

Environment Variables

Key:	Value:
GOROOT	/usr/local/go
GOPATH	//root/usr/local/go/src
PATH	//root/usr/local/go/src/bin:/usr/local/go...

```
package min

type Miner interface {
        Len() int
        ElemIx(ix int) interface{}
        Less(i, j int) bool
}

func Min(data Miner) interface{}  {

}

type IntArray []int
func (p IntArray) Len() int                       {}
func (p IntArray) ElemIx(ix int) interface{}  {}
func (p IntArray) Less(i, j int) bool             {}
```

```
type StringArray []string
func (p StringArray) Len() int                    {}

func (p StringArray) ElemIx(ix int) interface{}  {}
func (p StringArray) Less(i, j int) bool          {}
```

We hope that you were able to solve the challenge. The next lesson brings you the solution to this challenge.