



```
class User
{
    private $storage;

    public function __construct($storage)
    {
        $this->storage = $storage;
    }

    public function setLanguage(string $lang)
    {
        $this->storage->set('lang', $lang);
    }

    public function getLanguage():? string
    {
        return $this->storage->get('lang');
    }
}

$storage = new SessionStorage();
$user = new User($storage);
```



























































**А ЧТО ЕЩЕ ТУТ
МОЖНО УЛУЧШИТЬ?**

:)



ОЧЕНЬ ЛЕГКО КАСТОМИЗИРОВАТЬ

```
class User
{
    private $storage;

    public function __construct($storage)
    {
        $this->storage = $storage;
    }

    public function setLanguage(string $lang)
    {
        $this->storage->set('lang', $lang);
    }

    public function getLanguage():? string
    {
        return $this->storage->get('lang');
    }
}

$storage = new SessionStorage();
$user = new User($storage);
```

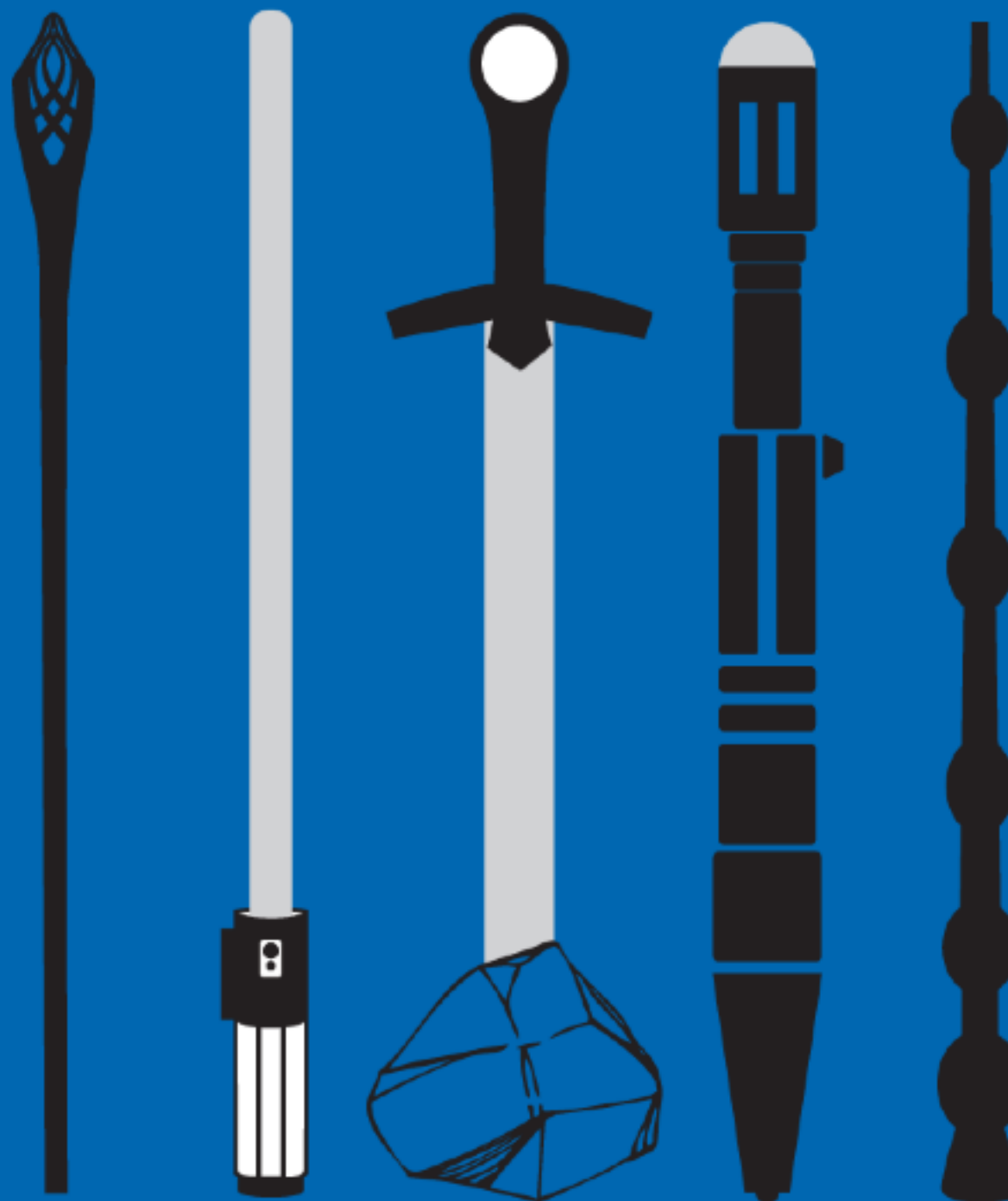
ЧУТЬ-ЧУТЬ СЛОЖНЕЕ В ИСПОЛЬЗОВАНИИ

А ЧТО ЕЩЕ ТУТ
МОЖНО УЛУЧШИТЬ?
:)



ИНТЕРФЕЙС

помогает нам регламентировать
скоуп возможностей объекта,
с которым мы работаем.



CHOOSE WISELY...