abstract class Model

{

public function StructCheckSet(){}

protected function SetAttributes(array $attributes\_data=[], bool $checkAttributeName = true){}

protected function GetAttributes(){}

protected function GetAttribute(string $attributeName){}

protected function SetFundamentalAttributes(int $id){}

public function \_\_toString(){}

public function \_\_call(string $methodName, $args){}

public function SetData(array $data){}

public function GetData(bool $attributes\_extra\_included, bool $attributes\_control\_included = false): array{}

public function GetClass(){}

public function GetClassName(){}

public function GetTablename(){}

}

class Zzz extends Model implements ModelInterface{

use ModelDatabase;

public function \_\_construct($id = 0){}

private function SetModelAttributes(){}

}

trait ModelDatabase{

load()

loadReferences()

save()

delete()

search()

length()

}

Abstract class Database