```
function clean_input() { }
private function save_dog_data()
{
if ( file_exists("e5dog_container.php")) {
                require_once("e5dog_container.php"); // use chapter 5 container w exception
                handling
        } else {
                throw new Exception("Dog container file missing or corrupt");
        }

        $container = new dog_container("dogdata"); // sets the tag name to look for in XML file
        $properties_array = array("dogdata"); // not used but must be passed into create_object
        $dog_data = $container->create_object($properties_array); // creates dog_data object
        $method_array = get_class_methods($dog_data);
        $last_position = count($method_array) - 1;
        $method_name = $method_array[$last_position];
        $record_Array = array(array('dog_name'=>"$this->dog_name", 'dog_weight'=>
        "$this->dog_weight", 'dog_color'=>"$this->dog_color", 'dog_breed'=>"$this->dog_breed"));
        $dog_data->$method_name("Insert",$record_Array);
        $dog_data = NULL;

}
function set_dog_name($value)
{
$error_message = TRUE;
(ctype_alpha($value) && strlen($value) <= 20) ? $this->dog_name = $value :
$this->error_message = FALSE;
return $this->error_message;
}
function set_dog_weight($value)
{
$error_message = TRUE;
(ctype_digit($value) && ($value > 0 && $value <= 120)) ? $this->dog_weight = $value :
$this->error_message = FALSE;
return $this->error_message; }
function set_dog_breed($value)
{
$error_message = TRUE;
($this->validator_breed($value) === TRUE) ? $this->dog_breed = $value :
$this->error_message = FALSE;
return $this->error_message; }
function set_dog_color($value)
{
$error_message = TRUE;
(ctype_alpha($value) && strlen($value) <= 15) ? $this->dog_color = $value :
$this->error_message = FALSE;
return $this->error_message;
}
```