1. **Which class is used to extend with Controller in Zend 1.11 and what is the purpose?**

**Ans.** Zend\_Controller\_Action is use for extend with each controller in Zend 1.11. It is a abstract class which we may implantation for action controller with front controller in MVC structure.

1. **What is init() action for controller in Zend 1.11 ?**

**Ans.** It calls at the initiation of controller each time.

1. **What is the Pre and Post-Dispatch hooks in Controller ?**

**Ans.** preDispatch() and postDispatch are the hooks for these. preDispatch() calls before each action of controller calling. We can use this to take decision before any action whether it should call or not. We can skipped the current action by using \_forward() in preDispatch().

1. **What is the difference b/w init() and preDispatch() ?**

**Ans.** Init() is generally use for extending the constructor. This works same as constructor of class. But preDispatch() use for taking decision before any action call whether it should call or not.

1. **How to get param \_GET or \_POST, explain with example ?**

**Ans.**

$id = $this->\_getParam('id', 1);

// Instead of:

if ($this->\_hasParam('id') {

$id = $this->\_getParam('id');

} else {

$id = 1;

}

1. **How to clear the SESSION of logged user in Zend 1.11?**

**Ans.**

$auth = Zend\_Auth::getInstance();

$auth->clearIdentity();

1. **How to get all the POST variable of form after submit in Zend 1.11 ?**

**Ans.** There is a function to get the POST variable of form after submit.

$formData = $this->getRequest()->getPost();

We can check is form has been submitted or not by this :

If($this->getRequest()->isPost()){}

1. **How we can check User is logged in or not in Zend 1.11 ?**

**Ans.**

We can do in this way :

$user = Zend\_Auth::getInstance();

If($user->hasIdentity()){}

If user is logged in then we can get the user information as object by this way :

$userinfo = $user->getStorage()->read();

1. **How to redirect to some other url in Zend 1.11 ?**

**Ans.**

Use this function :

$this->\_redirect($url);

1. **How to pass variables to view by controller in Zend 1.11 ?**

**Ans.**

In this way :

$this->view->content = $cont;

After this we can user $content in view file for showing content.

1. **How to call Model class in Controller in Zend 1.11 ?**

**Ans.**

In this way :

$login = new Application\_Model\_DbTable\_<Model Class name>();

1. **How to implement mail functionality in Zend 1.11 ?**

**Ans.**

In this way :

$mail = new Zend\_Mail(); // Initialization of mail class

$mail->setFrom(‘<from email>’,’<from title>’); // set from email

$mail->addTo(‘<to email>’,’<to title>’); // set to email

$mail->setSubject(‘<subject text>’); // set subject for mail

$mail->setBodyHtml(‘<message>’); // set html body

$mail->send(); // mail sending

We can clear addTo :

$mail->clearAddTo();

1. **How to use success, error messages in Zend 1.11 ?**

**Ans.**

In this way :

Success Message :

$this->\_helper->flashMessenger->addMessage(array(‘success’=>’<message>’));

Error Message :

$this->\_helper->flashMessenger->addMessage(array(‘error’=>’<message>’));

1. **How to get cookie in Controller in Zend 1.11 ?**

**Ans.**

In this way :

$cookie = Zend\_Controller\_Request\_Http::getCookie(‘<cookie name>’,’default’);

1. **How to authenticate user while login in Zend 1.11 ?**

**Ans.**

// Set table for user login info

$auth = new Zend\_Auth::getInstance();

$authAdapter = new Zend\_Auth\_Adapter\_DbTable();

$authAdapter->setTableName(‘<table name>’)->setIdentityColumn(‘username’)->setCredentialColumn(‘password’);

// get user login info from form

$username = $request->getParam(‘username’);

$password = $request->getParam(‘password’);

// set it identity for user

$authAdapter->setIdentity($username);

$authAdapter->setCredential(base64\_encode($password));

// check is it valid or not

$select = $authAdapter->getDbSelect();

$select->where(‘activestatus=1’);

$result = $auth->authenticate($authAdapter);

If($result->isValid())

{

$data = $authAdapter->getResultRowObject(null,’password’);

$auth->getStorage()->write($data);

}

1. **Name of the class which we uses for extending to Model ?**

**Ans.** Class name is Zend\_Db\_Table\_Abstract. We should use the protected variable $\_name for storing name of table.

We can declare the column for primary key for the current table defined in Model by using $\_primary variable.

1. **Can we use init() function in Model ?**

**Ans.** Yes, we can use this.

1. **How to insert values from Model to table in Zend 1.11 ?**

**Ans.** We need to prepare the array either in Model or while passing it to model and use this function $this->insert($data);

For example :

$data = array(

‘name’ => ’<value>’,

‘address’ => ‘<value>’,

);

$id = $this->insert($data);

$id is the last inserted id.

1. **How to update data in table in Zend 1.11 ?**

**Ans.**

We can user $this->update(‘<data array>’,’<where condition>’) function for this.

For example:

$this->update($data, ‘id=’.$id);

1. **How to delete from table in Model ?**

**Ans.**

In this way :

$this->delete(‘<where condition>’);

For example : $this->delete(‘id=’.$id);

1. **How to convert results in array in Model ?**

**Ans.** We use this function : $result->toArray();

1. **How to fetch all the results in Model ?**

**Ans.**

Use this function : $this->fetchAll(‘<where condition>’,’<column name>’);

1. **How to fetch a single row in Model ?**

**Ans.**

User this function : $this->fetchRow(‘<where condition>’);

1. **Which is the default Zend class use for extending our Form class ?**

**Ans.** Class name : Zend\_Form

1. **How to set method of form ?**

**Ans.**

In this way :

$this->setMethod(‘Post’);

1. **How to set action of form ?**

**Ans.**

In this way :

$this->setAction(‘<action url’);

1. **How to set attributes of form ?**

**Ans.**

In this way :

$this->setAttrib(‘id’,’<value>’);

$this->setAttrib(‘name’,’<value>’);

1. **How to add element in form to show in frontend ?**

**Ans.**

In this way :

$this->addElement(‘<input type>’,’<input name>’,array(‘label’=>’<value>’,’class’=>’<value>’,’required’=>true,’Description’=>’<value>’,’id’=>’<value>’, ‘style’=>’<value>’,’onclick’=>’<value>’));

1. **How to add element radio in Form ?**

**Ans.**

In this way :

$this->addElement(‘<input type>’, ‘<input name>’, array(‘label’=>’<value>’, ‘value’=>’<value>’, ‘multiOptions’=>array(1=>’Yes’,0=>’No’), ‘escape’=>true));

1. **What are the components of Zend ?**

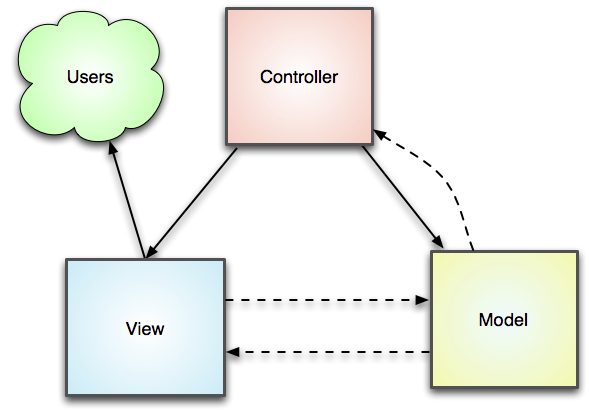
**Ans.**

Components are :

Zend\_Controller, Zend\_Layout, Zend\_Config, Zend\_Db, Zend\_Db\_Table, Zend\_Registry and along with a few View helpers.

1. **What is MVC ?**

**Ans.** Every application code falls under one of the following three categories: presentation, business login and data access. The MVC pattern models this separation of concerns well. Here are diagrammatically explanations:



 Model - This is the part of your application that defines its basic functionality behind a set of abstractions. Data access routines and some business logic can be defined in the model.

 View - Views define exactly what is presented to the user. Usually controllers pass data to each view to render in some format. Views will often collect data from the user, as well. This is where you're likely to find HTML markup in your MVC applications.

 Controller - Controllers bind the whole pattern together. They manipulate models, decide which view to display based on the user's request and other factors, pass along the data that each view will need, or hand off control to another controller entirely.

1. **What is the basic structure of Zend setup ?**

**Ans.**

1. |-- application
2. |   |-- Bootstrap.php
3. |   |-- configs
4. |   |   `-- application.ini
5. |   |-- controllers
6. |   |   |-- ErrorController.php
7. |   |   `-- IndexController.php
8. |   |-- models
9. |   `-- views
10. |       |-- helpers
11. |       `-- scripts
12. |           |-- error
13. |           |   `-- error.phtml
14. |           `-- index
15. |               `-- index.phtml
16. |-- library
17. |-- public
18. |   |-- .htaccess
19. |   `-- index.php
20. `-- tests
21. |-- application
22. |   `-- bootstrap.php
23. |-- library
24. |   `-- bootstrap.php
25. `-- phpunit.xml
26. **What are the design patterns in Zend ?**

**Ans.**

There are two design patterns in Zend :

* 1. Two Step View
  2. Composite View

*Two Step View* : the basic idea is that your application view creates a representation that is then injected into the master view for final transformation.

*Composite View* : deals with a view made of one or more atomic, application views.

1. **How to enable Layouts through Command and INI file ?**

**Ans.**

By Command : zf enable layout

By INI file :

resources.layout.layoutPath = APPLICATION\_PATH “/layouts/scripts”

1. **Which function works for initialization of Doctype in Bootstrap class and how ?**

**Ans.** \_initDoctype() is uses for initialization of Doctype in Bootstrap. Here is the example :

We need to initialize the view application.ini first with null like this :

*resources.view[] =*

After that we need to add the code in Bootstrap file :

*class Bootstrap extends Zend\_Application\_Bootstrap\_Bootstrap*

*{*

*protected function \_initDoctype()*

*{*

*$this->bootstrap('view');*

*$view = $this->getResource('view');*

*$view->doctype('XHTML1\_STRICT');*

*}*

*}*

Now we can use the doctype function in view file like this *<?php echo $this->doctype(); ?>*

1. **How to add stylesheets in layout file ?**

**Ans.**

Like this:

*<?php echo $this->headLink()->appendStylesheet(‘css/style.css’); ?>*

1. **How can we manage titles in Zend ?**

**Ans.**

We can set the title with separator with this function :

*$view = $this->getResource('View'); // set in bootstrap*

*$view->headTitle('My Site')->setSeparator(' :: '); // set in bootstrap*

*$this->headTitle()->prepend('Some Page'); // place before the set title*

1. **What is Decorators in Zend?**

**Ans.** Decorators uses for render elements and forms. It uses for represent elements and forms in the frontend. We can use this for form elements, errors, labels etc.

1. **How to manage SESSION in Zend?**

**Ans.**

Here is example :

*$mysession = new Zend\_Session\_Namespace('mysession');*

*if (!isset($mysession->counter)) {*

*$mysession->counter = 1000;*

*} else {*

*$mysession->counter++;*

*}*

*if ($mysession->counter > 1999) {*

*unset($mysession->counter);*

*}*

*The information stored in the above example is stored at $\_SESSION['mysession']['counter'].*

1. **How to add role, add resource and allow role to access particular resource?**

**Ans.**

For example :

*$acl = new Zend\_Acl();*

*// setup various roles by this*

*$acl->addRole(‘guest’);*

*// owner inherits all of the rules of guest*

*$acl->addRole(‘owner’, ‘guest’);*

*// adding resource in acl*

*$acl->addResource(‘blog’);*

*// add previllages*

*$acl->allow(‘guest’, ‘blog’, ‘view’);*

*$acl->allow(‘guest’, ‘blog’, ‘post’);*

*// check whether a role has get the permission to access*

*$acl->isAllowed($guestUser, $post, 'view');*

1. **What is Zend\_Acl ?**

**Ans.** (Access Control List) It’s a component which handles the task of building a tree of roles, resources and privileges to manage and query authorization requests against. Zend\_Acl provides a lightweight and flexible access control list (ACL) implementation for privileges management.

1. **Which library is use for pagination?**

**Ans.** Zend\_Paginator use for pagination in Zend.

1. **How to remove access controls from any resource ?**

**Ans.**

*$acl = new Zend\_Acl();*

*$acl->removeDeny(‘staff’,’latest’,’revise’);*

1. **What is Zend\_Amf ?**

**Ans.** Zend\_Amf is Action message format, to allow communication between adobe flash player and PHP. Specifically, it provides a gateway server implementation for handling request sent from flash player to server and mapping these requests to object and class methods and arbitrary callbacks.

1. **What is Application Controller?**

**Ans.** It is the default of Zend structure. It helps to run the application first time after setup. It set the PHP environment for Zend application.

1. **What is Frontend Controller?**

**Ans.** Its purpose is to initialize the request environment. It takes the incoming request and and convert it into action of the controller. It takes the request and send them back after completion of process of the same request.

1. **How to configure the Zend application with Database?**

**Ans.**

We need to add the configuration in application.ini. Here is code :

[production]

*resource.db.adapter = ‘pdo\_mysql’*

*resource.db.params.host = ‘localhost’*

*resource.db.params.username = ‘root’*

*resource.db.params.password = ‘xxxxxxx’*

*resource.db.params.dbname = ‘test’*

*resource.db.isDefaultTableAdapter = true*

1. **How to set multiple Db with Zend Application?**

**Ans.**

[production]

*resources.multidb.db1.adapter = "pdo\_mysql"*

*resources.multidb.db1.host = "localhost"*

*resources.multidb.db1.username = "webuser"*

*resources.multidb.db1.password = "XXXX"*

*resources.multidb.db1.dbname = "db1"*

*resources.multidb.db2.adapter = "pdo\_pgsql"*

*resources.multidb.db2.host = "example.com"*

*resources.multidb.db2.username = "dba"*

*resources.multidb.db2.password = "notthatpublic"*

*resources.multidb.db2.dbname = "db2"*

*resources.multidb.db2.default = true*

Retrieving a specific database adapter

*$resource = $bootstrap->getPluginResource('multidb');*

*$db1 = $resource->getDb('db1');*

*$db2 = $resource->getDb('db2');*

*$defaultDb = $resource->getDb();*

1. **What is Zend\_Auth Adapter?**

**Ans.** A Zend\_Auth adapter is used to authenticate against a particular type of authentication services such as LDAP, RDBMS or file based storage. For Zend\_Auth we generally used credentials for authentication with Database.