

**Benchmark Class**

mark(mark\_point)  
 elapsed\_time(start\_mark,end\_point)  
 memory\_usage()

**Calendaring Class**

generate(year,month,array)

**Shopping Cart Class**

insert()  
 update()  
 total()  
 total\_items()  
 contents()  
 has\_options(row\_id)  
 options(row\_id)  
 destroy()

**Config Class**

item(item\_name)  
 set\_item(item\_name,item\_value)

**Database Class - Queries**

query(select sql query)  
 simple\_query()  
 db\_prefix(prefix)  
 protect\_identifiers(table\_name,boolean)  
 escape(value)  
 escape\_str(value)

**Database Class - Query Results**

result()  
 result\_array()  
 row()  
 row\_array()  
 num\_rows()  
 num\_fields()  
 free\_result()

**Database Query Helper**

insert\_id()  
 affected\_rows()  
 count\_all(table\_name)  
 platform()  
 version()  
 last\_query()  
 insert\_string(table\_name,data\_array)  
 update\_string(table\_name,data\_array,where\_array)

**Database - Inserting Data**

insert(table\_name,data\_array)  
 set(field\_name,value)

**Database - Updating Data**

update(table\_name,data\_array,where\_array)

**Database - Deleting Data**

delete(table\_name,where\_array)  
 empty\_table(table\_name)  
 truncate()

**Database - Active Record Caching**

start\_cache()  
 stop\_cache()  
 flush\_cache()

**Transactions**

trans\_start()  
 trans\_complete()  
 trans\_status()  
 trans\_rollback()  
 trans\_commit()  
 trans\_strict()  
 trans\_off()  
 trans\_begin()

**Table Data**

list\_tables()  
 table\_exists(table\_name)

**Field Data**

list\_fields(table\_name)  
 field\_exists(field\_name,table\_name)  
 field\_data()

**Custom Function Calls**

call\_function(function\_name,param1,param2,etc.)

**Database Caching Class**

cache\_on()  
 cache\_off()  
 cache\_delete(controller,function)  
 cache\_delete\_all()

**Database Forge Class**

create\_database(db\_name)  
 drop\_database(db\_name)

**Creating and Dropping Tables**

add\_field(field\_array)  
 add\_key(field\_name,boolean)  
 create\_table(table\_name)  
 drop\_table(table\_name)  
 rename\_table(old\_name,new\_name)

**Modifying Tables**

add\_column(table\_name,fields\_array)  
 drop\_column(table\_name,column\_name)  
 modify\_column(table\_name,fields\_array)

**Database Utility Class**

list\_databases()  
 optimize\_table(table\_name)  
 repair\_table(table\_name)  
 optimize\_database()  
 csv\_from\_result(db\_result)  
 xml\_from\_result(db\_result)  
 backup()

**Email Class**

initialize(config\_array)  
 from(email\_address,name)  
 reply\_to(email\_address,name)  
 to(email\_address)  
 cc(email\_address)  
 bcc(email\_address)  
 subject(subject\_string)  
 message(message\_string)  
 set\_alt\_message(message\_string)  
 clear()  
 send()  
 attach(file\_path)  
 print\_debugger()

**Encryption Class**

encode(string,key)  
 decode(string)  
 set\_cipher(cipher\_type)  
 set\_mode(mcrypt\_mode)  
 sha(string)

**File Uploading Class**

do\_upload(field\_name)  
 display\_errors(open\_tag,close\_tag)  
 data()

**Form Validation**

set\_rules(array)  
 run()  
 set\_message(function\_name)

**Form Helper Reference**

form\_error(field\_name,open\_tag,close\_tag)  
 validation\_errors()  
 set\_value(field\_name)  
 set\_select(name,value,boolean)  
 set\_checkbox(name,value,boolean)  
 set\_radio(name,value,boolean)

**FTP Class**

connect(dbconfig\_array)  
 upload(local\_path,remote\_path,mode,permission)  
 rename(file\_path/name,new\_file\_path/name)  
 move(source\_path,destination\_path)  
 delete\_file(file\_path)  
 delete\_dir(dir\_path)  
 list\_files(dir\_path)  
 mirror(source\_dir\_path,new\_dir\_path)  
 mkdir(path/dir\_name,DIR\_WRITE\_MODE)  
 chmod(file\_path or dir\_path,DIR\_WRITE\_MODE)  
 close()

**HTML Table Class**

generate(data\_array or query)  
 set\_caption(name,other\_name,other\_name,etc.)  
 set\_heading(data\_array)  
 add\_row(value\_array)  
 make\_columns(data\_array,column\_length)  
 set\_template(array)  
 set\_empty(default\_value)  
 clear()

**Image Manipulation Class**

resize()  
 crop()  
 rotate()  
 watermark()  
 clear()  
 display\_errors(open\_tag,close\_tag)

**Input & Security Class**

xss\_clean(data)  
 post(form\_field\_name,boolean)  
 get(form\_field\_name,boolean)  
 get\_post(form\_field\_name,boolean)  
 cookie(cookie\_name,boolean)  
 server(item\_name,boolean)  
 ip\_address()  
 valid\_ip(ipaddress)  
 user\_agent()

**Loader Class**

view(file\_name,data\_array,boolean)  
 model(model\_name)  
 database(options,boolean)  
 scaffolding(table\_name)  
 vars(array)  
 helper(file\_name)  
 plugin(file\_name)  
 file(filepath/filename,boolean)  
 lang(file\_name)  
 config(file\_name)

**Language Class**

load(file\_name,language)  
 line(language\_key)

**Output Class**

set\_output(data)  
 get\_output()  
 set\_header(header\_string)  
 set\_status\_header(server\_status\_code)  
 enable\_profiler(boolean)  
 cache(no\_of\_seconds)

**Pagination Class**

initialize(config\_array)  
 create\_links()

**Session Class**

userdata(item)  
 set\_userdata(items\_array)  
 unset\_userdata(session\_name)  
 set\_flashdata(item,value)  
 flashdata(item)  
 keep\_flashdata(item)  
 sess\_destroy()

**Trackback Class**

send(trackback\_array)  
 display\_errors()  
 send\_error(string)  
 receive()  
 send\_success()  
 data(item)

**Template Parser Class**

parse(template\_name,data\_array,boolean)

**Typography Class**

auto\_typography(string)  
 format\_characters(string)  
 nl2br\_except\_pre(string)  
 protect\_braced\_quotes = boolean

**Unit Testing Class**

run(test,expected result,test name)  
 report()  
 result()  
 use\_strict(boolean)  
 set\_template(string)

**URI Class**

segment(segment\_no,default\_value)  
 rsegment(segment\_no,default\_value)  
 slash\_segment(segment\_no,slash\_postion)  
 slash\_rsegment(segment\_no,slash\_postion)  
 uri\_to\_assoc(segment\_no)  
 ruri\_to\_assoc(segment\_no)  
 assoc\_to\_uri(array)  
 uri\_string()  
 ruri\_string(segment\_no)  
 total\_segments()  
 total\_rsegments()  
 segment\_array()  
 rsegment\_array()

**User Agent Class**

is\_browser()  
 is\_mobile()  
 is\_robot()  
 is\_referral()  
 browser()  
 version()  
 mobile()  
 robot()  
 platform()  
 referrer()  
 agent\_string()  
 accept\_lang()  
 accept\_charset(character\_set)

**XML-RPC & XML-RPC Server Classes**

server()  
 timeout()  
 method()  
 request()  
 send\_request()  
 set\_debug(boolean)  
 display\_error()  
 display\_response()  
 send\_error\_message()  
 send\_response()

**Zip Encoding Class**

add\_data()  
 add\_dir()  
 read\_file()  
 read\_dir()  
 archive()  
 download()  
 get\_zip()  
 clear\_data()