《中间件》问答系统

版本：2.6

# 系统概述

玩家通过回答系统提供的题目获得额外的声望、金币等奖励；从问题库中随机选取；根据回答的问题不同，获得奖励不同；每天有一定的次数限制；每次答题有一定的时间间隔；

# 系统设计

答题中间件提供ExamQuestion抽象类负责抽取题目，验证答题是否正确，处理相应奖励的功能；

1. 选取题目

Extract方法：负责随机抽出相应的题目；

1. 答题

DoAnswer方法：负责验证答题是否正确，及处理奖励；

1. 获取奖励

GetPrize方法：返回处理奖励结果

# 使用示例

1. 扩展答题中间件

|  |
| --- |
| class DemoExamQuestion : ExamQuestion<QuestionData>  {  private object \_prizeObj;  public DemoExamQuestion(int userId)  : base(userId)  {  }  protected override object ProcessResult(int questionId, bool answerResult)  {  if (answerResult)  {  \_prizeObj = new ConfigCacheSet<QuestionInfo>().FindKey(questionId);  //处理奖励  }  return \_prizeObj;  }  public override object GetPrize()  {  return \_prizeObj;  }  } |

1. 选取题目

|  |
| --- |
| import clr, sys  from action import \*  from lang import Lang  clr.AddReference('ZyGames.Framework.Game')  from ZyGames.Framework.Game.Com.Exam import \*  from ZyGames.Framework.Game.Com.Model import \*  def takeAction(urlParam, parent):  actionResult = ActionResult()  user = parent.Current.User;  examquest = new DemoExamQuestion(user.UserId);  actionResult.List = examquest.Extract(1, m => false)  return actionResult |

1. 答题

|  |
| --- |
| def takeAction(urlParam, parent):  actionResult = ActionResult()  user = parent.Current.User;  examquest = new DemoExamQuestion(user.UserId);  examquest.DoAnswer (urlParam.questId, urlParam.answer)  actionResult.List = examquest. GetPrize()  return actionResult |