

VIRTUAL CAMPUS
--admin_number:integer -admins[*]:Administrator -prof_number:integer -proflist[*]:Professor -degree_number:integer -degreelist[*]:Degree -seminar_number:integer -seminalist[*]:Seminar -currentuser:User* -fdp_number:integer -projectlist:FDP*
+VirtualCampus() +~VirtualCampus() +addDegree() +addFDP() +addSeminar() +addCourse() +start() +run() +end()

DATE
-day:integer -month:integer -year:integer
+Date(d=1: integer, m=1: integer, y=1: integer) +Date(other: const Date&) +getday():integer +setday(d: Integer) +getmonth():integer +setmonth(m: integer) +getyear():integer +setyear(y: integer)

COURSE
-credits:integer -studentlist[*]: Link_us_res -degree: Degree* - teachers[2]: Link_us_res*
+Course (id="undefined":string, d=nullptr: Degree*, c=6:integer, s="unknown":string, t=nullptr: Link_us_res **) +Course (const Course&) +getcredits():integer +setcredits(c:integer) +addteacher(newteacher: Link_us_res*) +addStudent(newstudent: Link_us_res*)

ADMINISTRATOR
-mycampus:VirtualCampus* -createUser() -createResource() -createDeegree()
+Administrator(id="xxxxxx":string, vc=nullptr: VirtulCampus*) +Administrator (ident:string, admin: const Administrator&) +menu()

RESOURCE
#identification:string #status:string
+Resource(id="undefined":string, s="unknown":string) +Resource(const Resource&) + ~Resource() +setIdentification(id:string) +getIdentification():string +setStatus(s:string) +getStatus():string +addteacher(newteacher: Link_us_res*) +addstudent(newstudent: Link_us_res*)

MARK
-value:float -date_graded:Mark
+Mark(v=-1:float) +Mark(v:float, when:Date) +setvalue(v:float) +getvalue():float +setdate_graded(when:Date) +setdate_graded(day:integer, month:integer, year:integer) +getdate_graded:Date

PROFESSOR
#identifier:string #courselist[*]:Link_us_res #fdplist[*]:Link_us_res #seminarlist[*]:Link_us_res*
+Professor() + Professor(ident:string) +setidentifier(ident:string) +getidentifier:string + menu()

USER
+~ User() +menu() +getidentifier():string

LINK_US_RES
-userPtr: User -resourcePtr: Resource -partial[4]:Mark -final:Mark
+Link_us_res(u=nullptr: User *, r=nullptr:Resource *) +checkHealth() +connectCourseUser() +checkUserKind():bool

DEGREE
-name:string -student_number:integer -stulist: Student* -course_number:integer -courselist:Course*
+Degree() +Degree(n:string) +~Degree() +getname():string +searchStudentbyid(id:string): Student* +getstudent_number():integer +setname(n:string) +addCourse() +addStudent() +addFDP() +showstudents() +showcourses()

STUDENT
-SIN:integer -mydegree: const Degree* -mycourses[*]:Link_us_res -myseminars[*]:Link_us_res -myfdp: Link_us_res* -addCourse(link: Link_us_res *) -addSeminar(link: Link_us_res *) -addFDP(link: Link_us_res *)
+Student(sin=0:integer) +Student (sin:integer, d: Degree &) +Student(const Student &) +getSIN():integer +setSIN(s:integer) +getidentifier():string +enroll(res: Resource *) +menu()

SEMINAR
-coordinator: Link_us_res* -speaker:Link_us_res* -eventDate:Date -maxseats:integer -freeseats:integer -students: Link_us_res*
+Seminar(id="undefined":string, s="unknown":string, seatsValue=0:integer, spe=null:User*, coord=null:User*) +Seminar(id="undefined":string, s="unknown":string, seatsValue=0:integer, when:Date, spe=null:User*, coord=null:User*) +getmaxseats():const int +setmaxseats(s:int) +setdate(when:Date) +getdate():const Date setspeaker(spe:User *) getspeaker():const Link_us_res setcoordinator(coord:User *) getcoordinator():const Link_us_res addteacher(newteacher:Link_us_res *) addstudent(newstudent:Link_us_res *)

FDP
-tutor:Link_us_res* -co_tutor:Link_us_res* -student:Link_us_res*
+FDP(id="undefined":string, stu=nullptr: Link_us_res*, s="unknown":string, t=nullptr:Link_us_res*, c_t=nullptr:Link_us_res*) +FDP(other: const FDP&) +gettutor():const Link_us_res& +settutor(t:Link_us_res*) +getco_tutor():const Link_us_res& +setco_tutor(c_t:Link_us_res*) +getstudent():const Link_us_res& +setstudent(stu:Link_us_res*) +addteacher(newteacher:Link_us_res*) +addstudent(newstudent:Link_us_res*)