

অবজেক্ট ! অবজেক্ট ! !

আপনার মনেটো গাড়ির সাথে একটি ফেনা গাড়ির পার্থক্য কোথায়? অল্পে অল্পে তাই নয়! টয়োটা নির্দেশ যেনে হলে, জান বা যান, সামনে বা পিছনে ইচ্ছে তবু চালানো যায়, হেডলাইন খামোলা যায়, আবার ব্রেক দ্রুতও চালানো যায়, ফেনা গাড়ির এত সব গুণ নই! বেসেতা, ফেনা গাড়ির সাথে ও ডায়েরির একটি ফেনা জুড়ি নই, প্রয়োজনে একটি নিম্নেট কয়েল সিটম লেগ করুন। এবার ও মজিরনে আশ্চর্যকর, হেডলাইনটা এসব শুধু ফেনা যেনে পারে, কিন্তু গাড়িটো কার আর অস্বাভাবিক বা অস্বাভাবিক ক্ষমতা কি এক? গাড়ি নির্ভরনের ক্ষেত্রে মডেল, কালার, স্পিড, ক্যাপাসিটি এসব যথার্থে প্রশ্ন আমতে পারে। তাই হলে সেগুলোকে কখনোই ফেনা গাড়ির সঙ্গে তুলনা করা হয় না। কেন? প্রসঙ্গটিকে মজিরে লাভ নই, প্রোগ্রামিং-এর ক্ষেত্রে দুটি ফেনাটা এক।

আমি একটা বাক্য রসে যোগ দিতে চাই, বাস্তব নয়, মনবিরে। সমস্যা কোথায়? একই অবজেক্ট দুটা গাড়ি, একটা ফেনা, আরেকটা নীল, গাড়িরে গতি প্রতি ঘটায় ৫৫ মাইল, দ্বিতীয়টির ৫৩ মাইল। প্রচলিত প্রোগ্রামিং ল্যাঙ্গুয়েজের সঙ্গে যারা সুপরচিত, তারা খুব ভাল করেই জানেন, একেই দুটা গাড়িকেই আমরা অস্বাভাবিক সমঝাতি করতে হলে— তাদের চারটেই হইল, দুটো ছোট লাইট, অন্য দুটা বড় (কিছু বড়)। সবই দুজনে করে বলতে হবে। চালানোর সময়তো আরও বেশী। প্রতিটি গাড়িকে আলাদাভাবে একসময় থেকে আরেক স্থানে সরাতে হবে, প্রতিবারই নিউ আর পিচার নিয়ে মন্য ঘামতে হবে। ঠিক যেনে অস্থায়ী পর্যায়ে ফেনা গাড়ির মতো। কিন্তু এগুলি বই হতে— কন্সারের কিনতে পাওয়া হচ্ছে নানা মডেলের গাড়ি ফেনার কার গাড়ি ফেনার কার, সফটওয়্যারের বেসেটা) ইচ্ছে যত কিনে এনে মুছে দিলে আমরা প্রোগ্রামের সাথে, ফলস্বরূপ 'Start' সবগুলো গাড়ি পথে নমোনা। ১) অবজেক্টই আদিত কনসেপ্ট, কনসে প্রোগ্রামিং ল্যাঙ্গুয়েজ, ০,৪-এর মধ্যে নানা মডেলের প্রোগ্রাম পাওয়া যায়, কিন্তু এগুলোর ব্যবহার বা পরিচয় করা কি Start। করার মতই সম্বন্ধ? মোটেই নয়। এ থেকে কি সিদ্ধান্তে পৌঁছানো যায় যে 'প্রোগ্রামিং-এর ক্ষেত্রে আরও বড় জ্ঞান এক নয়? কিন্তু এ সিদ্ধান্তে এখন আর সত্য বলে মেনে নেয়া যায় না। এ যারা অভিজ্ঞ নয়, কিন্তু অনেক সামনে চলে এসেছে— OOP বা অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং (Object Oriented Programming) এর ফলা-কৌশল, প্রচলিত প্রোগ্রামিং-এর মূল-ধারার সঙ্গে তার রয়েছে ব্যাপক অমিল, কিন্তু বড় জগতের সঙ্গে যা পুরোপুরি সঙ্গাপূর্ণ।

যদি এক দশকের ম্যায়ার সিকে স্ট্যানফোর্ডের-এর SRI ইনস্টিটিউটের Doug Engelbart প্রত্যক্ষভাবে পরিচিনে ও নিয়ন্ত্রণযোগ্য নিম্নেট নিয়ে যে তথ্য ব্যক্তি করান তা আরও অনেক সিস্টেমেই ব্যবহৃত হচ্ছে। নমই-এর দশকের মাঝেই নমইয়েটে প্রথম অবজেক্ট অরিয়েন্টেটেড প্রোগ্রামিং ল্যাঙ্গুয়েজ, Simula-এর বিকাশ ঘটে। কিন্তু সমসাময়িককালে শক্তিপালী প্রোগ্রামিং ল্যাঙ্গুয়েজ C এর আবির্ভাব অবজেক্ট অরিয়েন্টেটেড প্রোগ্রামিং-এর উন্নয়নকে কিছুটা স্তিমিত করে। অবশেষে ১৯৮৩ সালে Bell Laboratories-এর Bjame Stroustrup C++ প্রোগ্রামিং-এর শক্তিপালী দিলকতলার সঙ্গে অবজেক্ট অরিয়েন্টেটেড প্রোগ্রামিংয়ের ধারণা বৃদ্ধ করে OOP-এর পালন নতুন ব্যবহার সম্ভার ঘটান। এই নতুন হাইব্রিড (Hybrid) জায়গার নাম হয়ে C++।

প্রত্যক্ষই বোঝা যায় অবজেক্ট (Object) কি? কোন একক সমস্যা অবজেক্টকে সমস্যাভিত করে যায় না। সাধারণভাবে বলা যায়, অবজেক্ট হল অবজেক্ট অরিয়েন্টেটেড সিস্টেমের পদ্ধতিগত একক, যা বড় জগতের কোন সত্তা বা এন্টিটি (Entity)-এর প্রতিকৃতি। বস্তু-সমূহ যেমন নিজস্ব পরিচিতি (Identity) আছে, সফটওয়্যারের অবজেক্ট-এরও তেমনি রয়েছে স্বকীয়তা। উদাহরণ হিসেবে বলা যেতে পারে গাড়ি, মানুষ, বস্তু মতই এক একটি সত্তা। দুটা গাড়ি একই মডেলের একই নির্ভাতার সৃষ্টি হওয়ায় তাদের গড়িত কার্যে একই গাড়ি বলা যায় না। দু'জনের আলাদা আলাদা জন্ম তারিখ, নাম ধাম একই হলেও দু'জন কিন্তু ভিন্ন মানুষ। তাদের গাড়িটো বৈশিষ্ট্য, আকার আত্মগা ভিন্নেই ন কিছু পার্থক্য থেকেই যায়। সময় ও স্থানের (Time and space)-এর বিভিন্ন তাসের আলাদাভাবে চিহ্নিত করা যায়। সময়ওস্থানের অবজেক্ট বড় জগতের এক স্বাভাবিক প্রতিচ্ছবি, এমনকি আটার আচরণ (Behavior) নই। আর তাই দু'জনের আলাদা থেকে দু'জনার আলাদা পরিলক্ষণ গড়ে তোলার মতই একটিই অবজেক্টের মতো একটি কার্যকরী সফটওয়্যার গড়ে তোলার ও নিয়ন্ত্রণ করা অসম্ভব হয়ে ওঠায় বেশ সহস্রসংখ্য। প্রতিটি সত্তারই যেমন কিছু বৈশিষ্ট্য রয়েছে, তেমনি প্রতিটি অবজেক্টেরও রয়েছে নিজস্ব বৈশিষ্ট্য বা Attributes। আর কাছ দিতে হলে দুটা মানুষকে আলাদা করা যায়, তেমনি দুটা অবজেক্টকেও

পৃথকভাবে চেনা যায়, অন্য অবজেক্টের সঙ্গে তাদের সম্পর্কের স্বরূপ থেকে বিভিন্ন পারস্পরিক সম্পর্কের ভিত্তিতে অবজেক্ট ভিত্তিভাবে কাজ করতে পারে; আর্টস্টার (Actors) যারা অন্য অবজেক্টের সাথে কাজ করে, হার্বার (Servers)-যারা অন্য অবজেক্ট কর্তৃক ব্যবহৃত হয়, এবং এজেন্ট (Agents) যারা অন্য অবজেক্টের পক্ষে কাজ করে।

সময়ের বেলাটা গাড়ির সঙ্গে যন্ত্র চালিত ফেনা গাড়ির তুলনা করে দেখুন— প্রথমটি শুধু ফেনা গাড়ি, ভাল শক্তি বহিষ্টি। চলতে চাইলে গাড়ির হাফের থেকে ধাক্কা দিতে হবে। কিন্তু দ্বিতীয়টি? চলার জন্যে প্রয়োজনীয় সিস্টেম উন্নয়ন ও উপকরণ সঙ্গে রয়েছে, শুধু চাই সৃষ্টিই অল করে সেয়া। ধারণ গাড়িটি হচ্ছে প্রচলিত প্রোগ্রামিং ল্যাঙ্গুয়েজের উপলক্ষে, যেমন ডাটা (Data) ও প্রক্রিয়া (Procedure) দুটা ভিন্ন যন্ত্র। অবজেক্টই এখন যে আপনি কোন কলি যা কলিগুলো তার প্রক্রিয়া বড় করা যায় কোন পথে নই (দানকতলা মেনে ফ্রেনের সঙ্গে মূর্খ পিঙ্গির তুলনাও করতে পারেন)। দ্বিতীয় গাড়ির চালিকা শক্তি রয়েছে গাড়ির বতির ভেতরেই অবজেক্ট অরিয়েন্টেটেড সিস্টেমের অবজেক্ট ও অনেকটা (পুরোপুরি নয়) যন্ত্রাঙ্গলিত ফেনা গাড়ির মত, ডাটা ও প্রক্রিয়ার সমন্বয়েই অবজেক্টের সৃষ্টি। ফলে নির্ধারিত আশানি একটি অবজেক্টের সঙ্গে আরেকটি অবজেক্টকে ছুড়ে দিতে পারেন বা নুনমত ব্যাফত খাটিয়ে একটি অবজেক্টকে ছুড়ে ফেলতে পারেন।

আরেকটু এনিমে দেখা যাক। আপনারা C++ কমপিউটারে অনেক মেরোয়ারের পরও এখন আরও ভাল করছেন। আপনি একটি 286 বা 386 মডেলের এনে। আপনি কি নতুন করে কমপিউটার বা ফেনা সফটওয়্যার ব্যবহার করা শিখেন? নিশ্চই না। গাড়ির ক্ষেত্রেও একই কথাই প্রযোজ্য— আপনি টয়োটার ব্যবহার জানলে সম্বোধেও চালানতে পারবেন। কিন্তু প্রচলিত প্রোগ্রামিং ল্যাঙ্গুয়েজের ক্ষেত্রে একথা বটে না। আমরা টোয়েন্ট ড্রাইভ প্রোগ্রামের একটি গাড়ির মডেল দেখাতে হলে, কিবা একটা মর সাইকেল ছুড়ে দিতে হলে আমাকে গাড়ির মডেল নমইতে কোন লিখেতে হবে, হরহাতের বা পুরো প্রোগ্রামটিই বাতিল হয়ে যাবে। অবজেক্ট ভিত্তিগত সিস্টেম এ জানা থেকে যত। প্রথম কারণটিই আবার বোলই— ডাটা ও প্রক্রিয়ার সমন্বয়েই। দ্বিতীয় কারণ হলে— আরেকটি সর্ব বহির্গতের যোগাযোগ পদ্ধতি (Method)।

অন্যায়ও ফেনা গাড়ির আলোচনাটা নিয়ে আসি। যন্ত্র চালিত গাড়িতে শক্তির প্রয়োজনীয় উপকরণ ট্রিকই আছে, কিন্তু তারপরও কি সে আপনার ইচ্ছে মত ডানে বামে ঘুরবে? মোটেই না। তার জন্যে চাই নিম্নেট কন্ট্রোল যার সাহায্যে অবজেক্ট প্রোগ্রামিং নির্দেশ পাঠাতে পারেন, গাড়ির চালককে নিয়ন্ত্রিত করতে পারেন। অবজেক্ট অরিয়েন্টেড সিস্টেম অবজেক্টের এনেই নিয়ন্ত্রণের জন্য ব্যবহৃত সংকেতের বা নির্দেশকে বলা হয় মেসেজ (Message)। প্রতিটি অবজেক্টই বহির্গত থেকে মেসেজের মাধ্যমে যোগাযোগ রাখে, তার ভেতরেই পঠন পঠন (Methods) সাহায্যে মিলিয়ে দেবে, উপযুক্ত কাজ করে বা, অপ্রয়োজনীয় সংকেত (যেমন Error Message) ফেরৎ পাঠায়। একই মেসেজ বিভিন্ন অবজেক্টে পাঠানোতে কোন চলিত্যতা নই। আর তাই আমি 'Start' বলে সবগুলো গাড়িকেই চলার নির্দেশ দিতে পারি, এবং উপযুক্ত অবজেক্ট নির্ভরন করে অবজেক্ট নমই 'Stop' বলে ধামতেও পারি; এ থেকে অবজেক্ট অরিয়েন্টেড সিস্টেমের আরেকটি উপলক্ষেরও যুক্ত পাওয়া যায়— OOP এর ভাষায় মর নাম এনক্যাপসুলেশন (Encapsulation)।

সমস্যা কাপসুলেশন করাই বিবেচনা করুন। ভিতরে কি আছে, তার জেনেরিক নাম কি, কখনো-কখনো ভিত্তি কি তা জানা কি আমরা জানা অপরিহার্য? কোন চিকিৎসকেরই তার বেলা ভাল রাখবে। কাপসুলেশন উপলক্ষে থেকে তার সমাধান নাম একই নাম সঞ্চালনা ন হওয়ার কোন লোভেই কিবা অসংক্রিপননে লেখা থাকে তার প্রয়োগ বিধি। গাড়ি বা কমপিউটারের ক্ষেত্রেও তাই। ইইংলার মানুষেরে লাবরণ ব্যবহারকারীদের জন্য প্রয়োজনীয় নির্দেশগুলো দিয়ে যেনা গকে— দুটি নাই টোবিনকাল টাচনের কুট-অমলে সেখানে নই। অবজেক্টেরও তেমনি ডাটা ও প্রক্রিয়া থাকে আনুত অবস্থার। কাপসুলেশন আরও হলে কিছু উন্মুক্ত তথ্য (Public Information) যেমন বিভিন্ন পদ্ধতির নামও সম্ভার লিখে তৈরী। এই পদ্ধতিগুলোর সঙ্গে যোগাযোগের মাধ্যম হল মেসেজ। এই পদ্ধতির হলে কখনোই ডাটা নই হওয়ার বা অন্য কোন পদ্ধতির মাধ্যমে বিবেক হওয়ার পর নই।

প্রচলিত প্রোগ্রামিং ল্যাঙ্গুয়েজে অভ্যস্ত প্রোগ্রামারদের পক্ষে অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়ের বহুক্ষেণ কার করার পথে একটি বড় বাধাই হল এই দুটি ভাষার ইয়কারীতা। প্রচলিত ল্যাঙ্গুয়েজের ভিত্তি হল "কোন ভাষার উপর ফাংশনের কিয়দ ঘটনো" যেমন কোন সমস্যাের বহুধন নিরিয়ে জন্য একটি ফাংশন লেখা যেতে পারে— Sqrt (X)। একেই ফাংশনিক X এর নাম তথা ভাষা চালিত সরবরাহ করতে হলে। সেখানেও সমস্যা— X যদি রিয়েল নাম্বারের বলা হয়, তাহলেই ইটোয়ার টাইপের ক্ষেত্রে এটি কার্য করবে না। অর্থাৎ 4.0 ও 4.0 এর জন্য এই পূর্বপত্র ফাংশনে, তাও আবার কোন নাম হতে পারবে না। অপ্রতিক্রিয়া OOP-এর ক্ষেত্রে বড় দর্শনই হলো— "অবজেক্টকে প্রয়োজনীয় সংকেত পাঠানো"। একেই অবজেক্টই হওয়া এক প্রয়োজনীয় সংকেত হতে পারে Sqrt বা বহুলি করার জন্য প্রয়োজনীয় নির্দেশ X.sqrt নির্দেশ এখানে X

অবশ্যেইকে বর্ণনুল নির্ণয়ের অন্য প্রয়োজনীয় নির্দেশ পাঠাবে, কি প্রক্রিয়ায় কাৰ্গটি সম্পন্ন করা হবে তা ক্যাপসুলের ভেতরেই রয়েছে। এখন যদি বর্ণনুল নির্ণয়ের ক্ষমতা কেবল খুঁজে পাওন যায়, তা হলে X অবশ্যেইটিকে পরিবর্তন করে নতুন অবশ্যেইটি হতে পারে। **Full information** এর কোন পরিবর্তন না করলেই হই।

আর ডাটা টাইপ নিয়ে যে আরওবর্তি ছিল, তাও ব্লক করার উপায় অবশ্যেইটির ক্ষেত্রেই রয়েছে। আমরা দেখাশাই এর আধুনিক দর্শনে 'আধুনিক' আধুনিক 'বাসে টিংকার খুঁজে দেখে যা, কারণ আমরা জানি আধুনিক বহুপলিমর্ফ থেকে পারে এবং বহুপলিমর্ফ (Polymorphism) অবশ্যেইটির অন্য বৈশিষ্ট্য। উদাহরণস্বরূপ 'আপারভেট অবশ্যেই ফোগ টিক ব্যবহার করে দুটি ইচ্ছাকৃতের ফেনে ফোগ করা থেকে পারে (1 + 2 = 3) তেমনই দুটি কার্যকরী শব্দকেও ফেন করা থেকে পারে ('abc' + 'def' = 'abcdef')। প্রস্তু উঠতে পারে, 'এ আর এমন কি? এ ডিবেক বা বেসিক নিজেও তো এ কাছ করা যায়। এ এক্ষেত্রে ডিবেক বা বেসিক বা কার, তা হচ্ছে সুপারবিসিয়াল, প্রকৃত প্রক্রিয়া এর মধ্যবর্তী পর্যায়ে রয়েছে। C বা Pascal এর মত টাইপিকিভাবে ভাষা প্রোগ্রাম লিখেন, তারা এ বিখ্যাত বহু সংশ্লেই অনুমান করতে পারবেন। অবশ্যেইটির আরিয়েটেড প্রোগ্রামিং-এ কাছগুলো বহু সংশ্লেই সম্পন্ন করা যায়। ডাটা প্রকৃতি উপর নির্ভর করে, অবশ্যেইটি নিজেই তার উপযুক্ত প্রক্রিয়াকে নির্ধারন করে।

এবার শেষ সমস্যার কথা বলি। চিহ্নটিটা সূচক দুটোই গাড়ি, ধরে নেই দুটোই চার চাকার গাড়ি, তাদের প্রয়োজনীয় স্থাপন রয়েছে, গতি রয়েছে, রয়েছে প্রয়োজনীয় সবকিছুই। এখন আমরা গাড়ি দুটো নিয়ে আলোচনা করব, তখন আমরা নিশ্চয়ই হেডারটি আর হুইলের আলোচনার ফিরে যাব না? এ যদি কোন বিশেষ বৈশিষ্ট্য নিয়ে আলোচনা করতে হয়, তবে সে ভিন্ন কথা। প্রোগ্রামিংয়ের সময় কিছু প্রক্রিয়াই আমাদের পুরো গাড়িটির কথা বিবেচনা করতে হবে, যদি দুটোর মডেল একই হয়, তাহলেও। তা যদি না করা হয়, তাহলে দেখা যাবে, গাড়ির বডি কেবল চাকাগুলোই শুধু বনিতের ছুটাছুটি করছে, আর প্রতিবারই আলোচনাতে হলে দেখা কম ফালেরা কাজ নয়। এর চেয়ে ভাল হয় যদি গাড়ি বলে দিলেই আমরা প্রোগ্রামটি বুঝে নেই, গাড়ির গতি প্রকৃতি থেকে অঙ্গসংস্থানে পর্যন্ত সবকিছুই। অবশ্য হই আর গতির পর্যক্য আমি আলোচনাতে বলে দিতে পারি। না বলে দিলে সে পূর্ণনির্ধারিত কিছু একটা আরোপ করে নিবে।

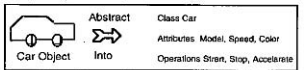
অন্যভাবেও বিশ্বয়কর উপস্থাপন করা যায় আমিও যানু, বনিও যানু, হাত-পা-মুখ-চেরখের বান্ধা উপস্থাপনার প্রয়োজন আছে কি? অবশ্যেইটির ক্ষেত্রেও তাই। একমিলক, একই প্রকার অবশ্যেইটি নিয়ে গড়ে উঠে একটি শ্রেণী (Class) আর এক শ্রেণীর বিভিন্ন সদস্য (Member বা instance) তাদের স্বকীয় কিছু বৈশিষ্ট্য (Unique Attributes) নিয়ে হয় পৃথকপৃথক। কিন্তু একেই সদস্যই তার শ্রেণীর স্বতন্ত্রতাকে কিছু বৈশিষ্ট্য ও আচরণ থেকে আঁকে। এ ধরনের বর্ণনাত্মকতা (Inheritance) হচ্ছে অবশ্যেইটির আরেকটি বৈশিষ্ট্য। ধীরে ধীরে ফেনে এই ধর্মটি সুপ্ত, তেমনই অবশ্যেইটির ক্ষেত্রেও। সুস্থভাবে রয়েছে হাঁস আর সন্ধ্যাক মিলে হাঁসসন্ধ্যাক হওয়ার মত ঘটনা ধীরে ধীরে প্রত্যাকভাবে না ঘটলেও পরোক্ষভাবে কি ঘটবে না? অবশ্যেইটি বা গাড়ী থাকবে কেন? একই অবশ্যেইটিও তাই একমিলক শ্রেণীর সদস্যে গঠিত হতে পারে, থাকে বলা হয় Multiple Inheritance. পূর্ববর্তী শ্রেণী বা শ্রেণীসমূহের সংশ্লেই এখানে বর্তমান থাকে, শুধুমাত্র নতুন ডাটা ও প্রক্রিয়াজ্ঞান একেই বলে দিতে হবে। এভাবেই আমি ফুটবলে তৈরী অত্যধিক ইচ্ছার সঙ্গে জাপানের তৈরী বডি মিলিয়ে আমরা বেসিকের টিকে অত্যধিক দ্রুত গতি সম্পন্ন ও সুদ্রুত করতে পারি।

এখানে বিভিন্ন প্রক্রিয়া (Procedure) ছাড়া ল্যাম্বার কাৰ্গটি দুভাবে হতে পারে। প্রথম পদ্ধতিকে বলে early binding, সেখানে প্রোগ্রাম তৈরীর সময়ই কোন প্রক্রিয়া কিভাবে সম্পন্ন হতে পারে, তা বলে দেয়া হয় এবং দ্বিতীয় পদ্ধতিকে বলে Dynamic বা Late binding, যেখানে কোন প্রক্রিয়ার সঙ্গে যুক্ত হতেওগুলো প্রোগ্রাম চলার সময়ই নির্ধারিত হয়। দ্বিতীয় পদ্ধতির একটি উদাহরণ হল মনিটর গাড়ির বিভিন্ন অংশগুলোকে অর্জন করা। যদি একটি প্রক্রিয়া হয় ব্লক অর্জন ও অপগার্ট হয় বহুভুক্ত অর্জন, তবে কোন কোন প্রক্রিয়াটি সম্পন্ন করতে হবে তা প্রোগ্রামটি চলানোলেই নির্ধারিত হবে। একইভাবে কোন ফাইল খ্রিট করার সময় ফাইলটি কি ধরনের ট্রেট ফাইল বা ডাটাবেক ফাইল, না কোন প্রক্রির ফাইল তা মূল ফাইলটি হতে পণ্ডরার পরই নির্ধারিত হবে এবং উপযুক্ত প্রক্রিয়াটি চালু হবে। প্রচলিত ল্যাম্বাফাংশনের সন সতর্ক থাকার প্রয়োজন নেই।

অবশ্যেইটির আরিয়েটেড ল্যাম্বাফাংশন এই বিভিন্ন বৈশিষ্ট্যগুলোর মূল দর্শনকে এক করার কথা হয় Data abstraction, এ যেখানে ডাটাই যুগ্ম, প্রক্রিয়া নির্ধারিত হয় ডাটার প্রয়োজনীয়তা-এর বিপরীতে প্রস্তুিত প্রোগ্রামিং ল্যাম্বাফাংশন-এর দর্শনকে বলা হয় Procedure Abstraction, যেখানে ফাংশন বা প্রক্রিয়ার উপযোগী ডাটা সরবরাহ করা হয়। ডাটা আনশ্রীকরণ হলো একটি 'ব্লক বস' যেখানে আভ্যন্তরীণ কাৰ্গগণাটি সম্পর্কে ধারণা পৌঁছ, সংকেত ও পদ্ধতি (যেখাঁ Message &

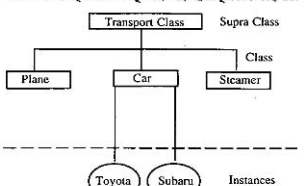
Method) যুগ্ম। ফলে ইন্ট্রিমেকশন তথা আভ্যন্তরীণ কাৰ্গগণাটি সম্পর্কে না জানেও অবশ্যেইকে পুরোপুরি ব্যবহার করা যায়। প্রোগ্রাম পরিবর্তন ও পরিষ্কারের কাছও হতে যায় অনেক সহজে। এবার তাহলে অবশ্যেইটির আরিয়েটেড প্রোগ্রামিং এর এই ধারণাগুলোকে টে-ট ডাউন প্রোগ্রামটি তৈরী করার কাছ পর্যায়ক্রমে প্রয়োগ করা যাক।

প্রোগ্রামটির মূল্যে কিছু গাড়ির প্রয়োজন। গাড়ি একটি বিশেষ শ্রেণীর অবশ্যেই, যা বহুভুক্ত গাড়ির গতি নির্দেশ, অর্থাৎ এর গতি, সংক্রান্ত ডাটা আছে, আর থামা, বলা গতি যাগুলো প্রকৃতি বক্রিমা আছে।



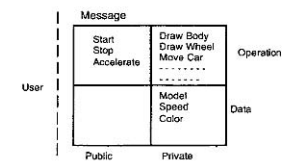
চিত্র ১: অবশ্যেই ও ক্লাস (Object & Classes)

প্রতিযোগিতার ব্যবহৃত বিভিন্ন গাড়িগুলোর এই গাড়ি শ্রেণীটির সদস্য এবং গাড়ি ট্রান্সপোর্টেড শ্রেণীর সদস্য ও ট্রান্সপোর্ট শ্রেণীর কিছু কিছু বৈশিষ্ট্য গাড়ির মধ্যে রয়েছে, যা সে উত্তরাধিকার মূলে লাভ করে। দুটাগাড়ি 'গাড়ি' শ্রেণীর দুটা অবশ্যেই।



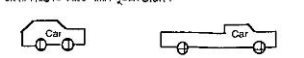
চিত্র ২: ইনহেরিটেন্স (Inheritance)

একটি গাড়ি ফেব্রানেই নির্মিত হোক না কেন, বিভিন্ন সংকেত বা পদ্ধতি ছাড়া থাকলে আমি গাড়িটিকে নিয়ন্ত্রণ করতে পারি, ডিভয়ের পরেই বা প্রক্রিয়া সম্পর্ক বিতৃত ধরনার প্রয়োজন নেই।



চিত্র ৩: এনক্যাপসুলেশন (Encapsulation)

যেহেতু গাড়ি একটি শ্রেণী, প্রয়োজনীয়তাও আমি এর মডেল, সংক্রান্ত পরিবর্তন করে আমরা প্রোগ্রামে যথেষ্ট সংখ্যক গাড়ি ব্যবহার করতে পারি (যেখাঁই কমপ্লিটগারেই মোডার্নেই খুব সংকুলন হতে হবে) একই সংকেত ব্যবহার করে ডাটার নিয়ন্ত্রণ করতে পারি। সূতরাং Start!



চিত্র ৪: ইন্সট্যান্সেস (Instances)

সম্পর্কে যাচির নতুন কোন অবজেক্ট (কোড) শেলে নির্দিষ্টায় পুরোনোটির পরিবর্তে নতুনটি ছুঁতে নিজে পরিচালিতের কোড ত্যাগি সেবিং নাই, নতুন কোডে আবার প্রোগ্রামের কিংবা আসে যায়।
ফুট নোট ১ :

- ১. বিট (Bit) : Binary digit ; বাইনারী পদ্ধতিতে ব্যবহৃত অক্ষর (০ বা ১)।
- ২. পিক্সেল (Pixel) : Picture element ; গ্রাফিক স্ক্রীনের একটি বিন্দু।

৩. প্রোগ্রামিং ভাষা/ভাষা (Programming Language) : কম্পিউটারকে কোন কাজ করার জন্য প্রোগ্রামিংয়ের নির্দেশ প্রদান করা। এ যন্ত্রের অন্যথা আবার প্রোগ্রামিংয়ের রয়েছে নিজস্ব পদ ভাষার বা ব্যাকব। যে ভাষায় ব্যবহৃত পদগুলো প্রায়, ইংরেজির মত, তাদের সারানাম নাম হইলেভলে ল্যাম্বুসে (High level বা English like language)। যেমন Pascal, C, BASIC, COBOL ইত্যাদি। কম্পিউটার কিংবা ইন্টারপ্রিটার ব্যবহার করে এদের কম্পিউটারের ভাষায় কাঙ্ক্ষিত। বাক্যে পদকে ইংরেজীতে লিখলে যে অর্থব্যয় দ্বারাও অনেকটা সেরকম। এখানে নিম্ন পর্যায়ের ভাষা (Low level language)। সবারনাম অ্যাসেম্বলি ব্যবহার করে এদের পুরোপুরি যেখান কোডে লিখারির করতে হয়। [C কে কখনও কখনও মধ্য পর্যায়ের ভাষা (Mid level language) হিসেবেও চিহ্নিত করা হয়]

৪. সোর্স কোড (Source code) : কোন প্রোগ্রামিং ভাষায় প্রোগ্রাম ব্যবহার করে কোন একটি কাজ সম্পন্ন করার জন্য প্রোগ্রামিংয়ের নির্দেশ। একটি কাজকে, প্রকারণ, ব্যবস্থাপনা এ ব্যবহারের সুবিধার্থে প্রায়ই ক্ষুদ্রতম কিছু অংশে বিভক্ত করা হয়। সবারনাম এর রুটিন (Routine), ফাংশন (Function) অথবা প্রসিডুরা বা প্রসিডুরি (Procedure) নামে পরিচিত। প্রতিটি ফাংশনে কোন নির্দিষ্ট টাইপের ডাটার উপর নির্দিষ্ট কিছু কাজ করে থাকে। এ যন্ত্রের কিছু যন্ত্রে এ তাদের পারস্পরিক সম্পর্কের সম্বন্ধে তৈরী হয় একটি কম্পিউটার প্রোগ্রাম। অবজেক্ট কোড (Object code) : সোর্স কোডের যেখান কোডে পরিণত হইল। আংশিক বা বিনা পরিবর্তনে এদের কম্পিউটারে ব্যবহার করা যায়। ইন্টার প্রিটার (যেমন GWBASIC) অবজেক্ট কোড তৈরী করে না।

৫. লাইব্রেরী (Program Library) : পরিচীকিত কিছু রুটিন বা ফাংশনের সমষ্টি। লাইব্রেরীর ব্যবহার এ ফাংশনগুলো পুনরাবৃত্তি করার জন্য প্রয়োজনীয় সময় এ শ্রম বিচার। নিরর্থকভাবে অনেক লাইব্রেরী সংগ্রহ এ ব্যবহারের মাধ্যমে প্রোগ্রামের দক্ষতা এ কার্যকরীতা বৃদ্ধি করে যায়।

৬. অবজেক্ট (Object) এ অবজেক্ট কোড (Object Code) : অবজেক্ট অসিমেট্রিক সিস্টেমের অবজেক্ট, অবজেক্ট কোডের (টিকা-০ দেখুন) সঙ্গে সম্পর্কিত নয়। সুব সাধারণত তাইয়া করা যায় অবজেক্ট ফাংশন এ ডাটার সম্বন্ধে। প্রোগ্রামিং ফেলে এটি সোর্স কোডের বিশেষ, অর্থ প্রোগ্রামিংয়ের উপকরণসমূহ বর্তমান। আপনার প্রোগ্রামের অন্য এটি সোর্স কোড বা অবজেক্ট কোড হইলে অবজারাইং সংগ্রহ বা ব্যবহার করিতে পারেন।

৭. ইন্টিগার (Integer) : পূর্ণ সংখ্যা ; যেমন ৫, ৭, ৩২ ইত্যাদি। রিয়েল (Real) : দশমিক সংখ্যা ; যেমন ২.৫, ০.৩০, ৪.০ ইত্যাদি। অক্ষর (Character) : একটি বর্ণ, সংকেত, চিহ্ন বা অক্ষর যেমন - A, B, C, \$, ^, 1 ইত্যাদি। স্ট্রিং (String) : কিছু ক্যারেক্টারের সমষ্টি। যেমন ABC, TK, 100.00 ইত্যাদি।

৮. Data Abstraction : "ডাটা বিমূর্তকরণ"।
 Procedure Abstraction : "প্রসিডুরা বিমূর্তকরণ"।

বালগোদেশ তত্ত্ব প্রযুক্তি আন্দোলনের পাবিক
কম্পিউটার জগৎ-এর সুলভ মূল্যের
কম্পিউটার বিষয়ক তথ্যবহুল গ্রন্থমালা

বেশিষ্টাসমূহ ১ : প্রতিটি বই তথ্যবহুল কিন্তু তথ্য জ্ঞানভর নয়। প্রাথমিক শিক্ষার্থী, অভিজ্ঞ ব্যবহারকারী সকলের জন্যই উপযোগী। পরিচালনা ছাড়া, সর্বদা ব্যবহার উপযোগী আয়তন।
 বিশেষ মূল্য ১- ধারা একত্রে কখনো ৪ টি বই কিনলে - ধারা বিশিষ্ট শিখা প্রতিষ্ঠানে শিক্ষার্থী - কম্পিউটার জগৎ-এর নিয়মিত গ্রাহককে (৪টি বই বিনামূল্যে পাঠ্য বই বিক্রয়) - কম্পিউটারের সাথে বিক্রয় প্রক্রিয়ায় - সকল উপাদান/শিখা প্রতিষ্ঠানসমূহ।

ডস সহায়িকা
 মূল্য : ৫০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

নেটাস সহায়িকা
 মূল্য : ৫০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

উইণ্ডোজ সহায়িকা
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

ওয়ার্ডস্টার সহায়িকা
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

ডিবেজ সহায়িকা
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

পিসি ট্রাবল শিট
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

ওয়ার্ডপারফেক্ট সহায়িকা
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

ডিটিপি সহায়িকা
 মূল্য : ৪০ টাকা
 বিশেষ মূল্য : ২৫ টাকা

গ্রন্থমালা প্রকাশ, দিনের ছুটি বই অন্যান্য আনিবার কারণে ছয় সংখ্যা কম্পিউটার জগৎ, আগামী ১৪ই জুন প্রকাশিত হচ্ছে। নিয়মিত গ্রাহকদের মধ্যে ধারা-এভাবে বইসমূহ পাননি তাদেরকে ছয় সংখ্যা কম্পিউটার জগৎ-এর সাথে ২৫ই জুন বৈশিষ্ট্য ডাকে/বইয়ে মাফত করে দেয়া হবে। আনুষঙ্গিক অনিয়ন্ত্রিত এ বিক্রয়ের জন্য পরিচয় পাঠকদের কাছে আনো পারিবে। আন্তরিকভাবে কৃতজ্ঞ। ছুটির সংখ্যা থেকে পরিকা যথারীতি মাফ প্রদান প্রকাশিত হবে।

প্রকাশক, **হাসিক কম্পিউটার জগৎ**

আপনার জন্য বিনামূল্যে কম্পিউটারের বই

আপনি কি কম্পিউটারের জগৎটাকে পুরোপুরি জানতে চান ?

তাহলে নিচের অল্পসুখ পুস্তক করে আবার **হাসিক কম্পিউটার জগৎ**-এর নিয়মিত গ্রাহক হোন এবং **কম্পিউটার জগৎ** প্রকাশনার বইসমূহ বিনামূল্যে পাবার সুযোগ গ্রহণ করুন।

নাম : _____

ঠিকানা : _____

কোন আস থেকে পত্রিকা পেতে ইচ্ছুক : _____

কোন কোন বই পেতে ইচ্ছুক : _____

গ্রাহক ঠাণ্ডা (রেজিষ্টার্ড ডাকে) এক বছরের জন্য দুই দশকট টাকা এবং ছয় মাসের জন্য এককল দশ টাকা চান ঠাণ্ডা মানি অর্ডার, ব্যালক ড্রাফট বা চেক নিচের ঠিকানায় পাঠিয়ে দিন। এক বছরের গ্রাহক দুইটি বই এবং ছয় মাসের গ্রাহক একটি বই বিনামূল্যে আপনার ঠিকানায় পাঠিয়ে দেয়া হবে।

হাসিক কম্পিউটার জগৎ

১৪৬/১, আজিমপুর মোড়, (চান্দা মিলিট এর গলি) ঢাকা-১২০৫। ফোন : ৫০৬৪৭৬ ফ্যাক্স : ৮৮০২-৮৬৬৭৪৬

আপনি যদি আগে থেকেই গ্রাহক হয়ে থাকেন, তবে এর একটি কপি আপনার কোন বন্ধু বা প্রিয়জনকে পুস্তক করে গ্রাহক হতে বলুন। এর সাথে আপনার নাম ও গ্রাহক নম্বর উল্লেখ করে, কোন বইটি বইটি আপনার পছন্দ তা আমাদের জানান। আপনার বন্ধু বা প্রিয়জন গ্রাহক হলে আপনিও তাদের দত বিনামূল্যে সমান সুযোগ লাভ করবেন।

কম্পিউটার জগৎ আপনার হাতের মুঠোয় থাকলে **কম্পিউটারের সমগ্র জগৎটাকে** আপনি জানতে পারবেন।